

คำนำ

หนังสือ การพัฒนาเว็บแอปพลิเคชัน เล่มนี้ จัดทำขึ้นเพื่อใช้เป็นตำราหลักของวิชา 4123309 วิชา การพัฒนาเว็บแอปพลิเคชัน (Web - based Application Development) ให้กับ นักศึกษาหลักสูตรวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราช ภัฏสวนดุสิต ซึ่งนักศึกษาจะได้เรียนรู้ถึงหลักการเขียนโปรแกรมเพื่อจัดการฐานข้อมูลบนเว็บ

การพัฒนาเว็บแอปพลิเคชันในปัจจุบันใช้เทคนิคการสร้างเว็บ ด้วยภาษาคอมพิวเตอร์ มากมาย โดยมีภาษาเอชทีเอ็มแอล เป็นภาษามาร์กอัปหลักในปัจจุบันที่ใช้ในการสร้างเว็บเพจ การแสดง ตัวอักษร รูปแบบการแสดงผล ตาราง หัวข้อ ลิงก์ ย่อหน้า รายการ รวมถึงการสร้าง แบบฟอร์ม ซึ่งนอกเหนือจากภาษา HTML ที่ใช้ในการแสดงผลในหน้าเว็บแล้ว ยังมี ภาษา สคริปต์ อีกมากมายที่ใช้ในการกำกับการทำงานต่าง ๆ เช่น ภาษาจาวาสคริปต์ ที่ใช้ในการ ตรวจสอบข้อมูล การแสดงข้อผิดพลาดที่เกิดขึ้น ซึ่งเรียกว่า เป็นภาษาในลักษณะไคลเอนต์-ไซด์ สคริปต์ แล้วนอกจากภาษาไคลเอนต์-ไซด์ สคริปต์ แล้วยังมีการเขียนโปรแกรมแบบเว็บเบส คือ มีการเก็บข้อมูลในต่าง ๆ ไว้ในฐานข้อมูลมีการเปลี่ยนแปลงข้อมูลตามกาลเวลา ตามผู้ใช้ ตาม ความต้องการของผู้ใช้แต่ละคนเรียกว่าไดนามิกเว็บ ก็จะมีภาษาคอมพิวเตอร์ในลักษณะ เซิร์ฟเวอร์-ไซด์ สคริปต์ เช่น ภาษาพีเอชพี PHP ในหนังสือเล่มนี้จะใช้ภาษาที่ใช้ในการพัฒนา ด้วย ภาษาพีเอชพี เป็นภาษาหลัก เพราะเป็นโอเพนซอร์ส ไม่มีปัญหาเรื่องลิขสิทธิ์ในการ พัฒนาต่อไปสำหรับการทำธุรกิจ

เนื้อหาในหนังสือเล่มนี้ ประกอบด้วย การเขียนโปรแกรม HTML การสร้างฟอร์ม FORM การเขียนโปรแกรม PHP และการประยุกต์ใช้งาน ฐานข้อมูล การเขียนโปรแกรม ติดต่อฐานข้อมูล เขียนโปรแกรม Session การเขียนกระดานข่าว การทำแบบสำรวจและการ สร้างสนทนา การนับจำนวนผู้เข้าชม ผู้เขียนได้แทรกคำอธิบายการทำงานของโปรแกรม และ ได้แสดงตัวอย่างที่สามารถใช้งานจริงได้มาแสดงให้เห็นอย่างชัดเจน สามารถนำไปประยุกต์ใช้ งานสร้างระบบสารสนเทศ ใช้งานได้จริง

นักสตรณย์ ชัชวาลานนท์

1 มิถุนายน 2555

สารบัญ

	หน้า
คำนำ.....	i
สารบัญ.....	ii
สารบัญตาราง.....	vi
สารบัญรูปภาพ.....	vii
บทที่ 1 การพัฒนาเว็บแอปพลิเคชัน Web - based Application Development.....	1
ภาษาพีเอชพี	1
การติดตั้งโปรแกรม AppServ.....	2
การรันโปรแกรม	7
บทที่ 2 การเขียนโปรแกรม HTML	9
ภาษา HTML.....	9
การสร้างตาราง	15
การสร้างเฟรม Frame.....	19
การสร้างฟอร์ม FORM	21
บทที่ 3 ภาษาจาวาสคริปต์.....	27
รูปแบบการเขียนโปรแกรม Java Script.....	27
การเขียนหมายเหตุในโปรแกรมภาษา Java Script.....	28
การประกาศตัวแปรของโปรแกรมภาษา Java Script	29
การรับค่าและการแสดงผลตัวแปร	30
การแปลงชนิดข้อมูล	32
ตัวดำเนินการ	33
การเขียนโปรแกรมแบบมีเงื่อนไข.....	35
เงื่อนไข 1 เงื่อนไข มีทางเลือก 2 ทาง.....	35
เงื่อนไขมากกว่า 1 เงื่อนไข	36
ตัดสินใจเลือกมากกว่า 2 ทางเลือก โดยใช้ switch-case	37
การเขียนโปรแกรมแบบวนซ้ำ.....	39
การวนซ้ำด้วยการใช้ while	41

สารบัญ(ต่อ)

การวนซ้ำด้วยการใช้ do-while	42
การสร้างฟังก์ชัน	42
การเรียกใช้งานฟังก์ชัน	43
บทที่ 4 การเขียนโปรแกรม PHP	47
รูปแบบการเขียนโปรแกรม PHP	47
การเขียนหมายเหตุในโปรแกรมภาษา PHP	48
คำสั่งพื้นฐานของโปรแกรมภาษา PHP	48
คำสั่งแสดงผล.....	49
ตัวดำเนินการทางคณิตศาสตร์	50
ตัวดำเนินการเพิ่มค่าและลดค่า	52
การเขียนโปรแกรมแบบมีเงื่อนไข.....	54
เงื่อนไข 1 เงื่อนไข มีทางเลือก 2 ทาง.....	54
เงื่อนไขมากกว่า 1 เงื่อนไข	55
ตัวดำเนินการเปรียบเทียบ.....	56
switch-case : ตัดสินใจเลือกมากกว่า 2 ทางเลือก.....	57
การเขียนโปรแกรมแบบวนซ้ำ.....	59
การวนซ้ำด้วยการใช้ while	61
การวนซ้ำด้วยการใช้ do-while	61
การประยุกต์ใช้งาน.....	62
บทที่ 5 ฐานข้อมูล	71
ฐานข้อมูลและการจัดการฐานข้อมูล (Databases and Databank Management)	71
ประเภทของการออกแบบฐานข้อมูล.....	73
แนวโน้มของฐานข้อมูล	75
การสร้างฐานข้อมูล MySQL	77
การสร้างตาราง	77
การตั้งค่า ภาษาไทย	81

สารบัญ(ต่อ)

การสร้างแบบสอบถาม SQL Query	81
บทที่ 6 การเขียนโปรแกรมติดต่อฐานข้อมูล	95
การแสดงผลข้อมูลจากฐานข้อมูล	95
การติดต่อ SERVER.....	96
การติดต่อฐานข้อมูล	97
การรันคำสั่ง SQL	98
การดึงข้อมูลมาแสดง.....	99
การดึงข้อมูลมาแสดงแบบวนซ้ำ	100
การแสดงผลข้อมูลเป็นตาราง	101
การแสดงรูปแบบตัวเลข	102
การจัดรูปแบบข้อมูล	102
การใส่หัวข้อตาราง	103
การแสดงสี่สลับสี.....	104
การเพิ่มข้อมูลในฐานข้อมูล.....	105
การแก้ไขข้อมูล	107
การลบข้อมูลในฐานข้อมูล	111
บทที่ 7 การใช้ตัวแปร Session.....	113
การลงทะเบียนตัวแปร Session.....	113
การทำลายตัวแปร Session.....	114
ระบบสมาชิก	115
ฐานข้อมูลเพื่อเก็บข้อมูลสมาชิก.....	115
บทที่ 8 การสร้างกระดานข่าว	137
ฐานข้อมูลกระดานข่าว	137
การแก้ไขปัญหาภาษาไทย.....	141
การสร้างคำถาม	141
การแสดงผลคำถาม	145

สารบัญ(ต่อ)

การตอบคำถาม	147
บทที่ 9 การทำแบบสำรวจ	151
ฐานข้อมูล poll_question	151
ฐานข้อมูล poll_answer	151
การสร้างห้องสนทนา	158
การนับจำนวนผู้เข้าชม	163
บทที่ 10 การประยุกต์ใช้งาน	167
บรรณานุกรม	170

สารบัญตาราง

หน้า

ตารางที่ 3-1	ตัวดำเนินการทางคณิตศาสตร์.....	33
ตารางที่ 3-2	ตัวดำเนินการทางคณิตศาสตร์แบบลดรูป.....	33
ตารางที่ 3-3	แสดงการเพิ่มค่า และ การลดค่า	34
ตารางที่ 3-4	แสดงตัวดำเนินการเปรียบเทียบ	34
ตารางที่ 4-1	ตัวดำเนินการทางคณิตศาสตร์	51
ตารางที่ 4-2	ตัวดำเนินการทางคณิตศาสตร์แบบลดรูป.....	51
ตารางที่ 4-3	แสดงการเพิ่มค่า และ การลดค่า	53
ตารางที่ 4-4	แสดงตัวดำเนินการเปรียบเทียบ	57

สารบัญรูปภาพ

หน้า

รูปที่ 1-1 : โปรแกรม AppServ สำหรับการติดตั้ง	3
รูปที่ 1-2 : ผลลัพธ์การรันโปรแกรม AppServ.....	3
รูปที่ 1-3 : หน้าจอ ยอมรับข้อตกลง.....	4
รูปที่ 1-4 : หน้าจอ การเลือกโฟลเดอร์	4
รูปที่ 1-5 : หน้าจอการใส่ชื่อ Server และ Email	5
รูปที่ 1-6 : หน้าจอการใส่รหัสผ่าน ของ root	5
รูปที่ 1-7 : หน้าจอขณะติดตั้งโปรแกรม	6
รูปที่ 1-8 : หน้าจอติดตั้งโปรแกรมเสร็จ	6
รูปที่ 1-9 : ผลลัพธ์การทดสอบ AppServ	7
รูปที่ 1-10 : ผลลัพธ์การรันโปรแกรม Hello World.....	8
รูปที่ 2-1 : ผลลัพธ์การแสดงชื่อเว็บเพจ.....	10
รูปที่ 2-2 : ผลลัพธ์การใส่สีพื้นและสีอักษร	11
รูปที่ 2-3 : ผลลัพธ์การใส่หัวข้อ	12
รูปที่ 2-4 : ผลลัพธ์การกำหนดหัวข้อ	12
รูปที่ 2-5 : ผลลัพธ์การสร้างตาราง	16
รูปที่ 2-6 : ผลลัพธ์การสร้างตารางและรูป.....	18
รูปที่ 2-7 : ผลลัพธ์การสร้างเฟรม.....	20
รูปที่ 2-8: ผลลัพธ์การสร้างเฟรม 4 เฟรม	21
รูปที่ 3-1 : ผังการไหลคำสั่ง if แบบ 2 ทางเลือก	35
รูปที่ 3-2 : ผังการไหลคำสั่ง if แบบหลายทางเลือก	36
รูปที่ 3-3 : ผังการไหลคำสั่ง switch – case	38
รูปที่ 3-4 : ผังการไหลคำสั่ง for	39
รูปที่ 3-5 : ผลลัพธ์ การใช้คำสั่ง for	40
รูปที่ 3-6 : ผลลัพธ์ การใช้คำสั่ง for โดยเพิ่มครั้งละ 2	40
รูปที่ 3-7 : ผังการไหลคำสั่ง while	41

สารบัญรูปภาพ(ต่อ)

รูปที่ 4-1 : ผลลัพธ์การแสดงผล	50
รูปที่ 4-2 : ผังการไหลคำสั่ง if แบบ 2 ทางเลือก	54
รูปที่ 4-3 : ผังการไหลคำสั่ง if แบบหลายทางเลือก	55
รูปที่ 4-4 : ผังการไหลคำสั่ง switch – case	57
รูปที่ 4-5 : ผังการไหลคำสั่ง for	59
รูปที่ 4-6 : ผลลัพธ์ การใช้คำสั่ง for	60
รูปที่ 4-7 : ผลลัพธ์ การใช้คำสั่ง for โดยเพิ่มครั้งละ 2	60
รูปที่ 4-8 : ผังการไหลคำสั่ง while	61
รูปที่ 4-9 : แสดงผลลัพธ์การรันโปรแกรม QUIZ.....	69
รูปที่ 5-1 : แสดงหน้าจอฐานข้อมูล MySQL	77
รูปที่ 5-2: แสดงหน้าจอสร้างฐานข้อมูล	78
รูปที่ 5-3 : แสดงหน้าจอสร้างชื่อตาราง.....	78
รูปที่ 5-4 : แสดงหน้าจอสร้างฟิลด์.....	79
รูปที่ 5-5 : แสดงหน้าจอสร้างฐานข้อมูล OA.....	79
รูปที่ 5-6 : แสดงหน้าจอสร้างตารางพนักงาน	79
รูปที่ 5-7 : แสดงหน้าจอการกำหนดรายละเอียดตารางพนักงาน	80
รูปที่ 5-8 : แสดงหน้าจอผลการสร้างตารางพนักงาน	80
รูปที่ 5-9 : แสดงหน้าจอผลการสร้างตารางพนักงาน	80
รูปที่ 5-10 : แสดงหน้าจอการตั้งคำรหัสภาษาไทย	81
รูปที่ 5-11 : แสดงหน้าจอการสร้างแบบสอบถาม	82
รูปที่ 5-12 : แบบสอบถามแสดงพนักงานแผนกคอมพิวเตอร์	83
รูปที่ 5-13 : แบบสอบถามแสดงพนักงานที่เงินเดือนมากกว่า 20000	84
รูปที่ 5-14 : แบบสอบถามแสดงพนักงานแผนกบัญชีที่เงินเดือนมากกว่า 20000	84
รูปที่ 5-15 : แบบสอบถามแสดงพนักงานแผนกบัญชีกับแผนกคอมพิวเตอร์.....	85
รูปที่ 5-16 : แสดงการใช้ Between	86
รูปที่ 5-17 : แสดงการใช้ เครื่องหมายไม่เท่ากับกับตัวเลข	86

สารบัญรูปภาพ(ต่อ)

รูปที่ 5-18 : แสดงการใช้ เครื่องหมายไม่เท่ากับกับข้อความ	87
รูปที่ 5-19 : แสดงการใช้ เครื่องหมาย LIKE	88
รูปที่ 5-20 : แสดงการใช้ เครื่องหมาย NOT LIKE.....	88
รูปที่ 5-21 : แสดงการใช้ IN.....	89
รูปที่ 5-22 : แสดงการใช้ %	90
รูปที่ 5-23 : แสดงการใช้ Order By.....	90
รูปที่ 5-24 : แสดงการใช้ คำสั่ง INSERT	91
รูปที่ 5-25 : แสดงการใช้ คำสั่ง Update	91
รูปที่ 5-26 : แสดงการใช้ คำสั่ง Update แบบมีเงื่อนไข	92
รูปที่ 5-27 : แสดงการใช้ คำสั่ง DELETE.....	93
รูปที่ 6-1 : แสดงตารางข้อมูลฟิลด์ในตารางพนักงาน.....	95
รูปที่ 6-2 : แสดงชื่อไฟล์ตารางพนักงาน	96
รูปที่ 6-3 : แสดงการติดต่อเครื่องแม่ข่าย	96
รูปที่ 6-4 : แสดงการติดต่อฐานข้อมูล	97
รูปที่ 6-5 : แสดงการแสดงผลเป็นตาราง	101
รูปที่ 6-6 : แสดงการแสดงผลเป็นตารางแบบสลับสี	104
รูปที่ 6-7 : แสดงการคีย์ข้อมูลเพื่อเพิ่มข้อมูล	105
รูปที่ 6-8 : แสดงการเพิ่มข้อมูลในตาราง	106
รูปที่ 6-9 : การแสดงผลพร้อมการเชื่อมต่อ.....	108
รูปที่ 6-10 : แสดงการแก้ไขข้อมูลในตาราง	109
รูปที่ 6-11 : แสดงผลลัพธ์การลบข้อมูล	112
รูปที่ 7-1 : แสดงข้อมูลรายละเอียดสมาชิก	116
รูปที่ 7-2 : แสดงหน้าเข้าสู่ระบบ.....	116
รูปที่ 7-3 : แสดงผลลัพธ์เข้าสู่ระบบไม่สำเร็จ	119
รูปที่ 7-4 : แสดงผลลัพธ์เข้าสู่ระบบได้	119
รูปที่ 7-5 : แสดงผลลัพธ์การรหัสผู้ใช้ไม่ถูกต้อง	124

สารบัญรูปภาพ(ต่อ)

รูปที่ 7-6 : แสดงผลลัพธ์การรหัสผู้เช่า.....	124
รูปที่ 7-7 : แสดงหน้าจอกรณีลืมรหัสผ่าน	126
รูปที่ 8-1 : แสดงรายละเอียดตาราง webboard_question	137
รูปที่ 8-2 : แสดงรายละเอียดตาราง webboard_answer	137
รูปที่ 8-3 : แสดงผลลัพธ์การแสดงผลภาษาไทย.....	140
รูปที่ 8-4 : แสดงการตั้งคำถามใหม่	142
รูปที่ 8-5 : แสดงผลลัพธ์การตั้งคำถามใหม่	144
รูปที่ 8-6 : แสดงผลลัพธ์การแสดงผลหัวข้อคำถาม.....	144
รูปที่ 8-7 : แสดงการตอบคำถาม	147
รูปที่ 8-8 : แสดงผลลัพธ์การตอบคำถาม	148
รูปที่ 8-9 : แสดงผลลัพธ์การแสดงผลจำนวนคำตอบของคำถาม	149
รูปที่ 8-10 : แสดงความคิดเห็นและคำตอบ	150
รูปที่ 9-1 : แสดงรายละเอียดตาราง poll_question	151
รูปที่ 9-2 : แสดงรายละเอียดตาราง poll_answer	151
รูปที่ 9-3 : แสดงผลลัพธ์การเพิ่มข้อมูล	155
รูปที่ 9-4 : แสดงการเลือกโหวต	156
รูปที่ 9-5 : แสดงผลการโหวต	158
รูปที่ 9-6 : แสดงผลการสนทนา	163
รูปที่ 9-7 : แสดงไฟล์ Counter.txt.....	164
รูปที่ 9-8 : แสดงผลลัพธ์การนับจำนวนผู้ชม	165
รูปที่ 9-9 : แสดงไฟล์ภาพในการแสดงตัวเลข	165

บทที่ 1 การพัฒนาเว็บแอปพลิเคชัน

Web - based Application Development

การพัฒนาเว็บแอปพลิเคชันในปัจจุบันใช้เทคนิคการสร้างเว็บ ด้วยภาษาคอมพิวเตอร์มากมาย โดยมีภาษาเอชทีเอ็มแอล HTML (เอชทีเอ็มแอล - เป็นคำย่อจากคำขึ้นต้นของ Hypertext Markup Language) เป็นภาษามาร์กอัปหลักในปัจจุบันที่ใช้ในการสร้างเว็บเพจ หรือข้อมูลอื่นที่เรียกดูผ่านทางเว็บเบราว์เซอร์ ซึ่งตัวโค้ดจะแสดงโครงสร้างของข้อมูล ในการแสดงตัวอักษร รูปแบบการแสดงผล ตาราง หัวข้อ ลิงก์ ย่อหน้า รายการ รวมถึงการสร้างแบบฟอร์ม ซึ่งนอกเหนือจากภาษา HTML ที่ใช้ในการแสดงผลในหน้าเว็บแล้ว ยังมี ภาษา สคริปต์ อีกมากมายที่ใช้ในการกำกับการทำงานต่าง ๆ เช่น ภาษาจาวาสคริปต์ ที่ใช้ในการตรวจสอบข้อมูล การแสดงข้อผิดพลาดที่เกิดขึ้น ซึ่งเรียกว่า เป็นภาษาในลักษณะไคลเอนต์-ไซด์ สคริปต์ แล้ว นอกจากภาษาไคลเอนต์-ไซด์ สคริปต์ แล้วยังมีการเขียนโปรแกรมแบบเว็บเบส คือมีการเก็บข้อมูลในต่าง ๆ ไว้ในฐานข้อมูลมีการเปลี่ยนแปลงข้อมูลตามกาลเวลา ตามผู้ใช้ ตามความต้องการของผู้ใช้แต่ละคนเรียกว่าไดนามิกเว็บ ก็จะมีภาษาคอมพิวเตอร์ในลักษณะเซิร์ฟเวอร์-ไซด์ สคริปต์ เช่น ภาษาพีเอชพี PHP ย่อมาจากคำว่า PHP Hypertext Preprocessor หรือชื่อเดิม Personal Home Page ในหนังสือเล่มนี้จะใช้ภาษาที่ใช้ในการพัฒนาด้วย ภาษาพีเอชพี เป็นภาษาหลัก เพราะเป็นโอเพนซอร์ส ไม่มีปัญหาเรื่องลิขสิทธิ์ในการพัฒนาต่อไปสำหรับการทำธุรกิจ

ภาษาพีเอชพี

ภาษาพีเอชพี (wikipedia.org) คือ ภาษาคอมพิวเตอร์ในลักษณะเซิร์ฟเวอร์-ไซด์ สคริปต์ โดยลิขสิทธิ์อยู่ในลักษณะโอเพนซอร์ส ภาษาพีเอชพีถูกคิดค้นขึ้นในปีค.ศ.1994 (พ.ศ. 2537) โดย Mr.Rasmus Lerdorf เพื่อใช้ตรวจสอบสถิติการเข้าชมเว็บของตนเอง ต่อมาในปีค.ศ. 1995 มีการเผยแพร่ภาษานี้สู่สาธารณะ และพัฒนาเป็น PHP/FI หรือพีเอชพีรุ่น 2 จนกระทั่งกลางปีค.ศ.1996 เริ่มมีทีมพัฒนาอย่างจริงจังโดยมี Mr.Zeev Suraski และ Mr.Andi Gutmans ร่วมทีม และพัฒนาเป็นพีเอชพีรุ่น 3

ภาษาพีเอชพีใช้สำหรับจัดทำเว็บไซต์ และแสดงผลออกมาในรูปแบบ HTML โดยมีรากฐานโครงสร้างคำสั่งมาจากภาษา ภาษาซี ภาษาจาวา และ ภาษาเพิร์ล ซึ่ง ภาษาพีเอชพี นั้นง่ายต่อการเรียนรู้ ซึ่งเป้าหมายหลักของภาษานี้ คือให้นักพัฒนาเว็บไซต์สามารถเขียน เว็บเพจ ที่มีความตอบโต้ได้อย่างรวดเร็ว

การแสดงผลของพีเอชพี จะปรากฏในลักษณะHTML ซึ่งจะไม่แสดงคำสั่งที่ผู้ใช้เขียน ซึ่งเป็นลักษณะเด่นที่พีเอชพีแตกต่างจากภาษาในลักษณะไคลเอนต์-ไซด์ สคริปต์ เช่น ภาษาจาวาสคริปต์ ที่ผู้ชมเว็บไซต์สามารถอ่าน ดูและคัดลอกคำสั่งไปใช้เองได้ นอกจากนี้พีเอชพียังเป็นภาษาที่เรียนรู้และเริ่มต้นได้ไม่ยาก โดยมีเครื่องมือช่วยเหลือและคู่มือที่สามารถหาอ่านได้ฟรีบนอินเทอร์เน็ต ความสามารถในการประมวลผลหลักของพีเอชพี ได้แก่ การสร้างเนื้อหาอัตโนมัติ จัดการคำสั่ง การอ่านข้อมูลจากผู้ใช้และประมวลผล การอ่านข้อมูลจากตาต้าเบส ความสามารถจัดการกับคุกกี้ ซึ่งทำงานเช่นเดียวกับโปรแกรมในลักษณะCGI คุณสมบัติอื่นเช่น การประมวลผลตามบรรทัดคำสั่ง (command line scripting) ทำให้ผู้เขียนโปรแกรมสร้างสคริปต์พีเอชพี ทำงานผ่านพีเอชพี พาร์เซอร์ (PHP parser) โดยไม่ต้องผ่านเซิร์ฟเวอร์หรือเบราว์เซอร์ ซึ่งมีลักษณะเหมือนกับ Cron (ใน ยูนิกซ์หรือลินุกซ์) หรือ Task Scheduler (ในวินโดวส์) สคริปต์เหล่านี้สามารถนำไปใช้ในแบบ Simple text processing tasks ได้

การแสดงผลของพีเอชพี ถึงแม้ว่าจุดประสงค์หลักใช้ในการแสดงผล HTML แต่ยังสามารถสร้าง XHTML หรือ XML ได้ นอกจากนี้สามารถทำงานร่วมกับคำสั่งเสริมต่างๆ ซึ่งสามารถแสดงผลข้อมูลหลัก PDF แฟลช (โดยใช้ libswf และ Ming) พีเอชพีมีความสามารถอย่างมากในการทำงานเป็นประมวลผลข้อความ จาก POSIX Extended หรือ รูปแบบ Perl ทั่วไป เพื่อแปลงเป็นเอกสาร XML ในการแปลงและเข้าสู่เอกสาร XML เรารองรับมาตรฐาน SAX และ DOM สามารถใช้รูปแบบ XSLT ของเราเพื่อแปลงเอกสาร XML

เมื่อใช้พีเอชพีในการทำอีคอมเมิร์ซ สามารถทำงานร่วมกับโปรแกรมอื่น เช่น Cybercash payment, CyberMUT, VeriSign Payflow Pro และ C CVS functions เพื่อใช้ในการสร้างโปรแกรมทำธุรกรรมทางการเงิน

การติดตั้งโปรแกรม AppServ

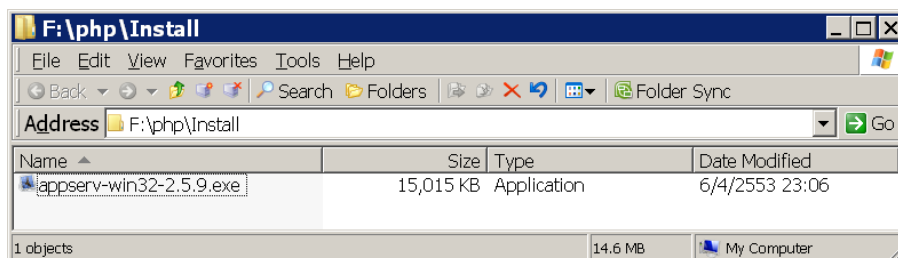
การติดตั้งโปรแกรม AppServ เพื่อจำลองเครื่องคอมพิวเตอร์ส่วนบุคคล เป็นเครื่อง Server เพื่อใช้ในการทดสอบ โปรแกรม PHP เสมือนทำงานบนเครือข่ายอินเทอร์เน็ตจริง โดยมีไฟล์เดอร์เพื่อเก็บข้อมูล คือที่ C:\AppServ\www โดยเมื่อติดตั้งโปรแกรม AppServ เรียบร้อยแล้ว ไฟล์เดอร์นี้ จะถูกสร้างขึ้นให้อัตโนมัติ และโปรแกรมต่าง ๆ ที่ต้องการรันผ่าน Server ให้บันทึกไว้ในไฟล์เดอร์นี้ เสมือนไว้ในเครื่อง Server

การติดตั้งโปรแกรม AppServ ให้ใช้ AppServ version 2.5.9 สำหรับ วินโดวส์ Windows XP โดยให้ดาวน์โหลด appserv-win32-2.5.9.exe สำหรับ วินโดวส์ Windows 7 ให้ใช้ AppServ version 2.5.10 โดยมีขั้นตอนการติดตั้งโปรแกรมเหมือนกัน คือมี 5 ขั้นตอน คือ

1. รันโปรแกรม appserv-win32-2.5.9.exe
2. ยอมรับข้อตกลง โดยกดปุ่ม I Agree
3. เลือกไฟล์เดอร์เพื่อบันทึกโปรแกรม โดยกำหนดที่ C:\AppServ แล้วกดปุ่ม Next

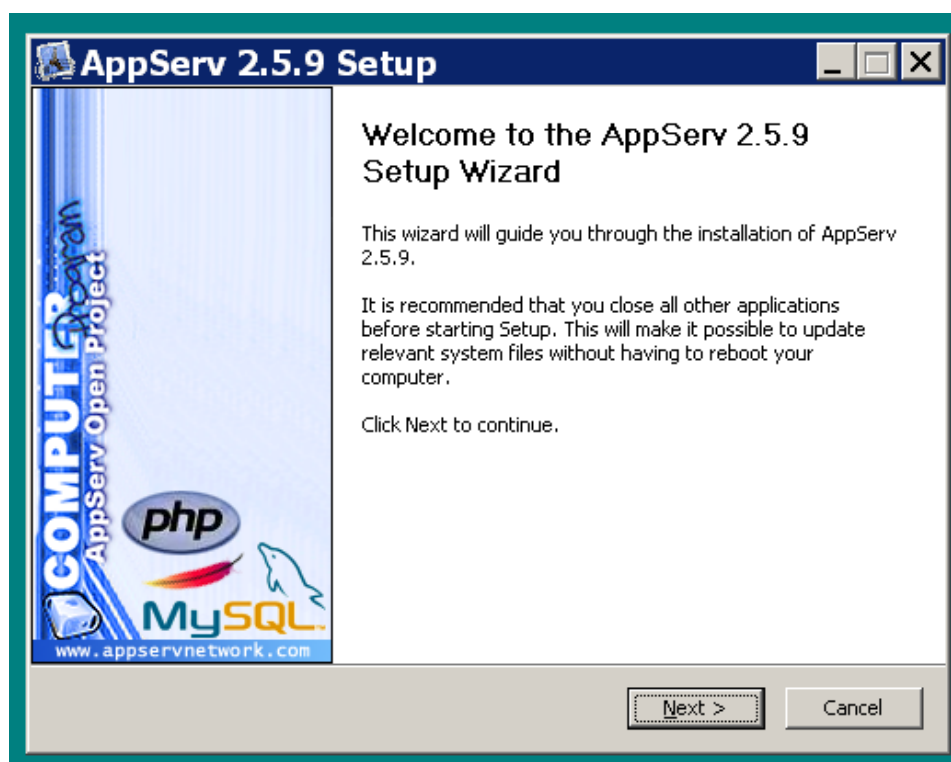
4. กำหนดชื่อเครื่องแม่ข่ายเป็น localhost และใส่ Email Address แล้วกดปุ่ม Next
5. กำหนดรหัสผ่าน เป็น 1234 โดยใส่ รหัสผ่าน 2 ครั้งให้ตรงกัน แล้วกดปุ่ม Install

ขั้นตอนที่ 1 รันโปรแกรม appserv-win32-2.5.9.exe



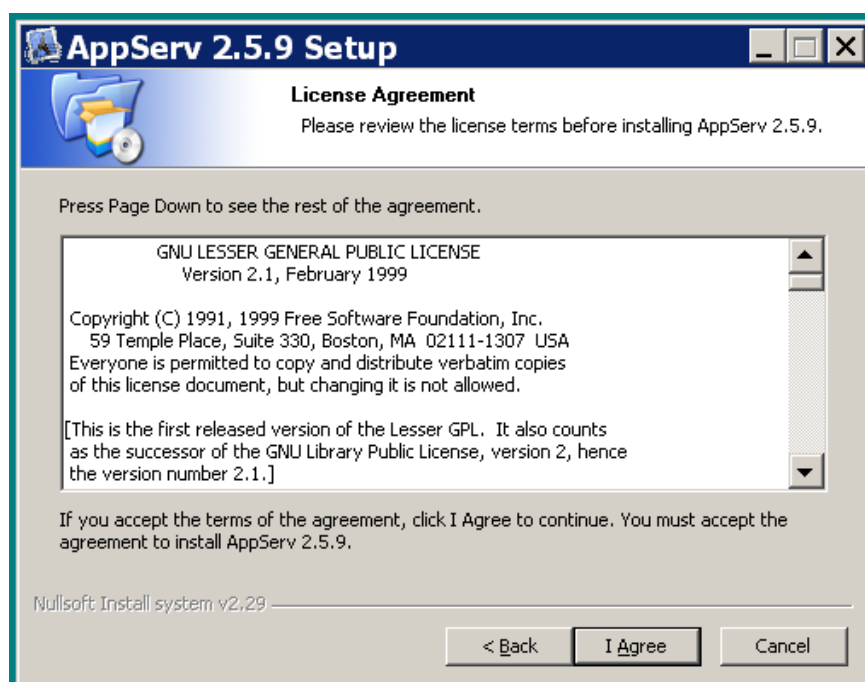
รูปที่ 1-1 : โปรแกรม AppServ สำหรับการติดตั้ง

ผลลัพธ์ การรันโปรแกรม appserv-win32-2.5.9.exe



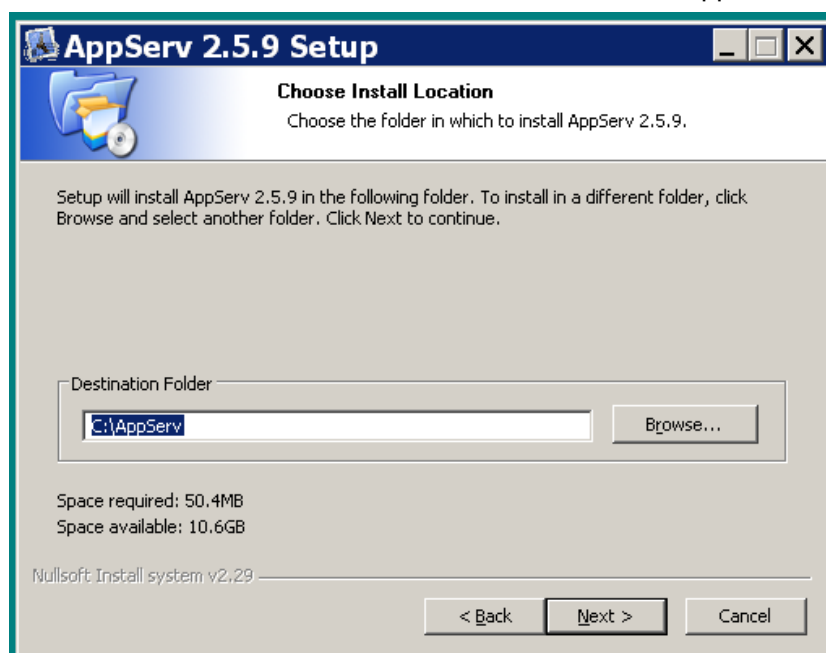
รูปที่ 1-2 : ผลลัพธ์การรันโปรแกรม AppServ

ขั้นตอนที่ 2 ยอมรับข้อตกลง โดยกดปุ่ม I Agree



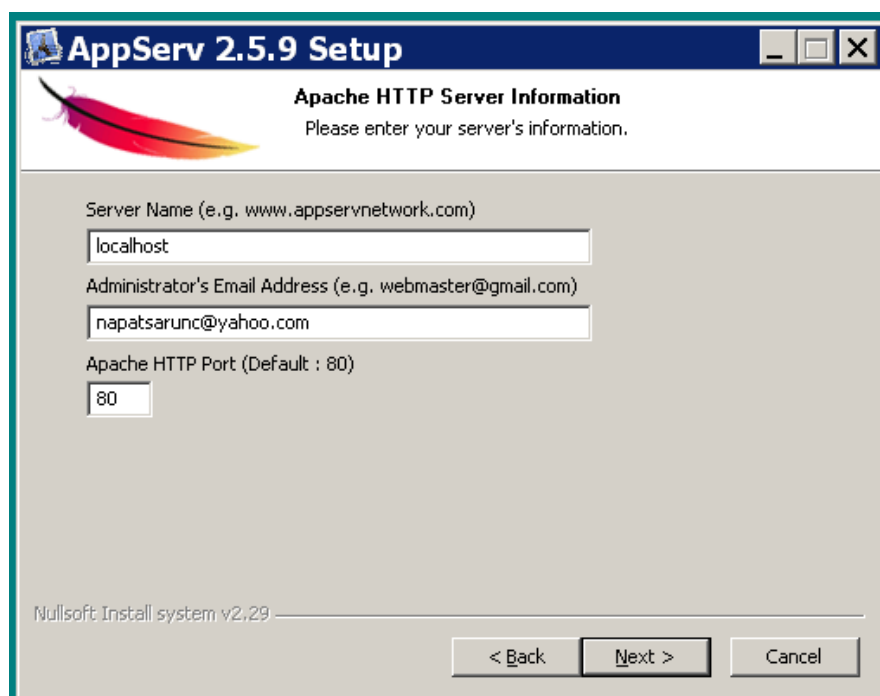
รูปที่ 1-3 : หน้าจอ ยอมรับข้อตกลง

ขั้นตอนที่ 3 เลือกโฟลเดอร์ เพื่อบันทึกโปรแกรม โดยกำหนดที่ C:\AppServ แล้วกดปุ่ม Next



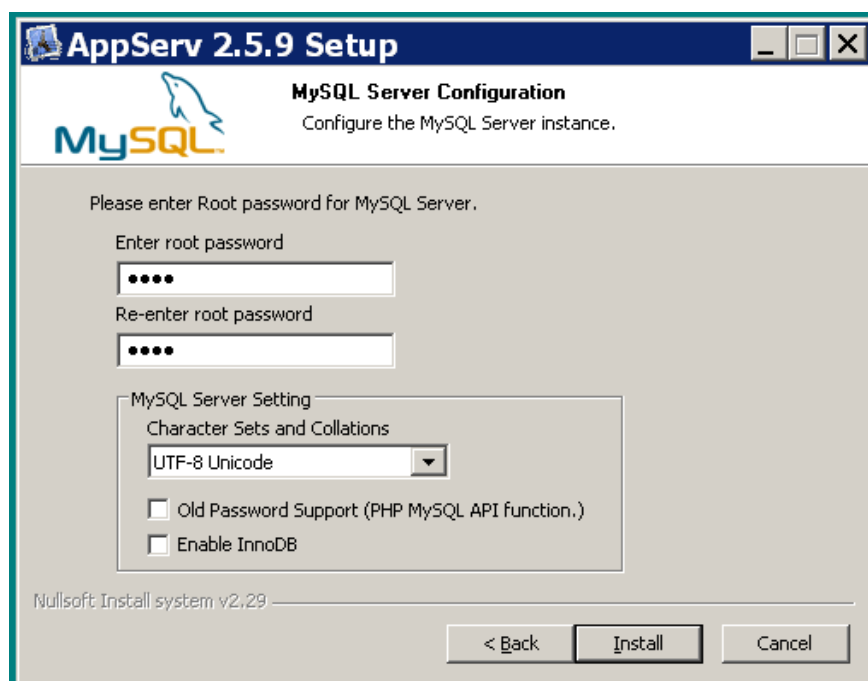
รูปที่ 1-4 : หน้าจอ การเลือกโฟลเดอร์

ขั้นตอนที่ 4 กำหนดชื่อเครื่องแม่ข่ายเป็น localhost และใส่ Email Address แล้วกดปุ่ม Next



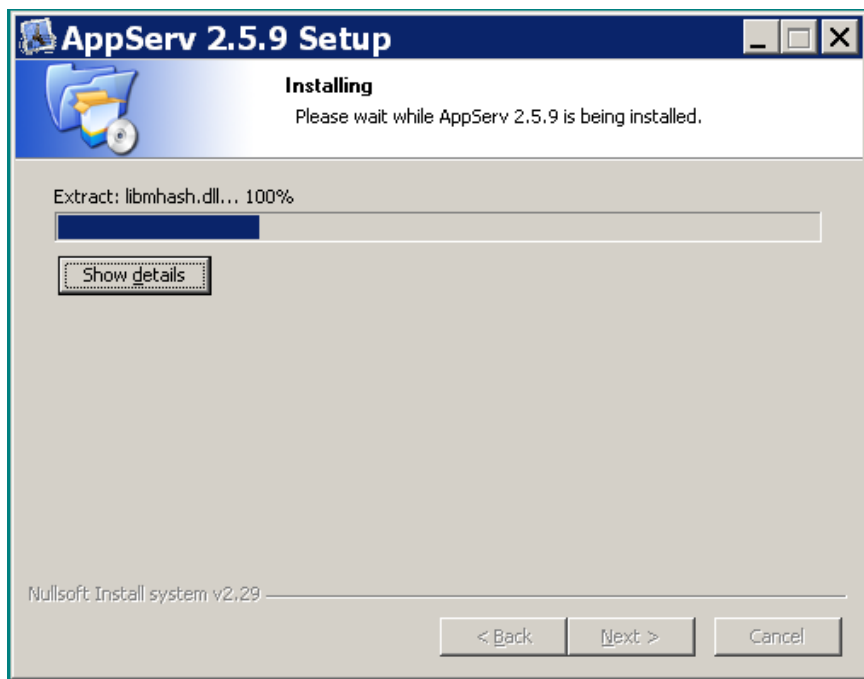
รูปที่ 1-5 : หน้าจอการใส่ชื่อ Server และ Email

ขั้นตอนที่ 5 กำหนดรหัสผ่าน เป็น 1234 โดยใส่ รหัสผ่าน 2 ครั้งให้ตรงกัน แล้วกดปุ่ม Install



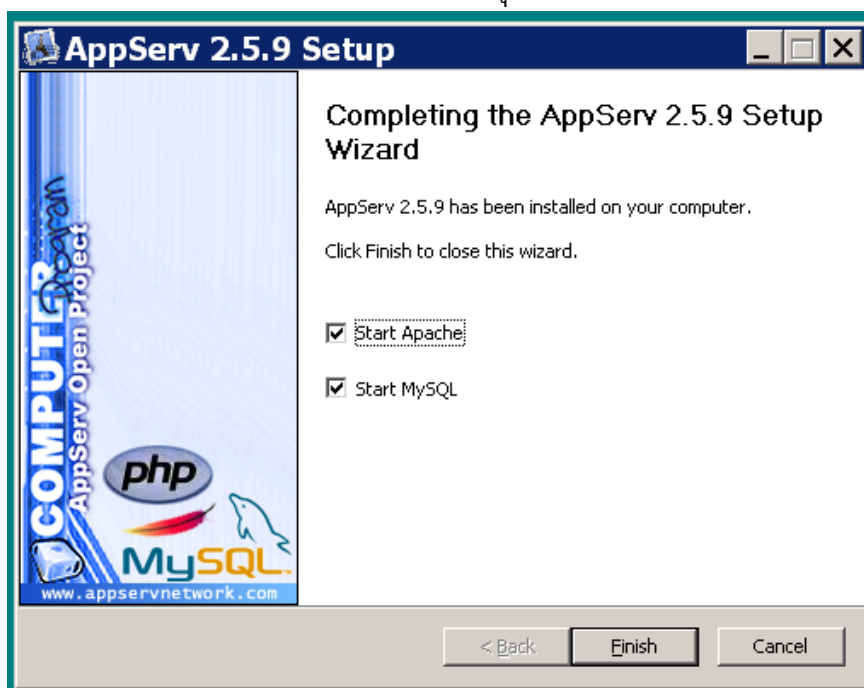
รูปที่ 1-6 : หน้าจอการใส่รหัสผ่าน ของ root

ผลลัพธ์ การลงโปรแกรม ให้รอจนโปรแกรมลงครบ 100 %



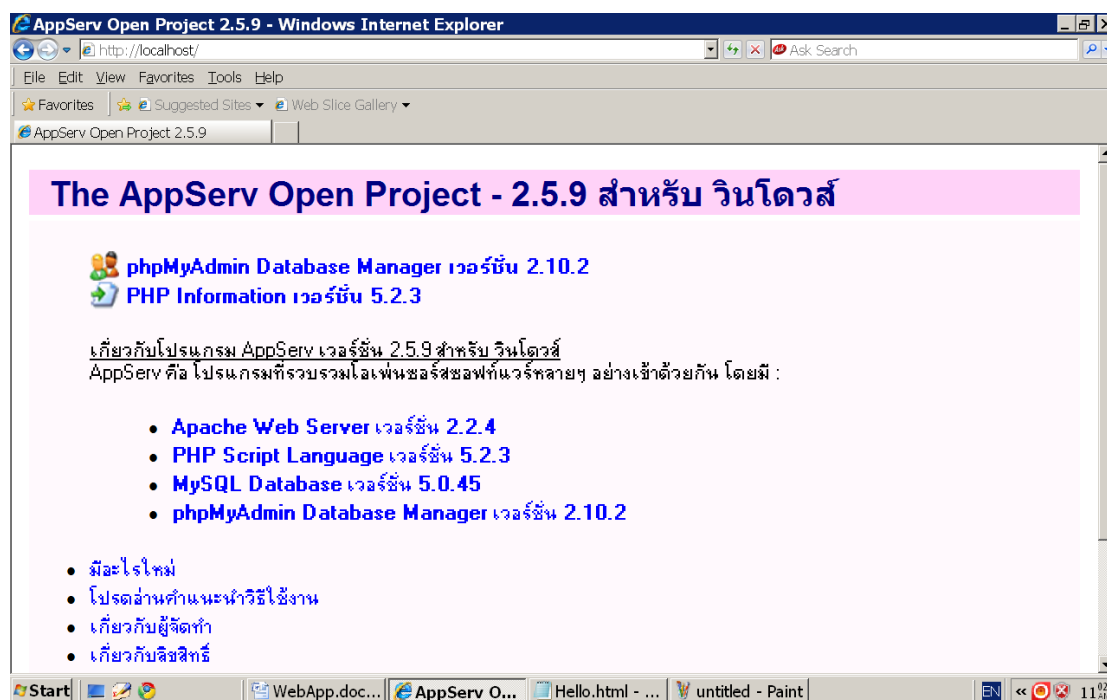
รูปที่ 1-7 : หน้าจอขณะติดตั้งโปรแกรม

ผลลัพธ์ เมื่อโปรแกรม ลงครบ 100 % กดปุ่ม Finish



รูปที่ 1-8 : หน้าจอติดตั้งโปรแกรมเสร็จ

เมื่อลงโปรแกรมแล้วให้ทดสอบด้วย <http://localhost> หากโปรแกรม AppServ รันได้ ผลลัพธ์จะเป็นดังนี้



รูปที่ 1-9 : ผลลัพธ์การทดสอบ AppServ

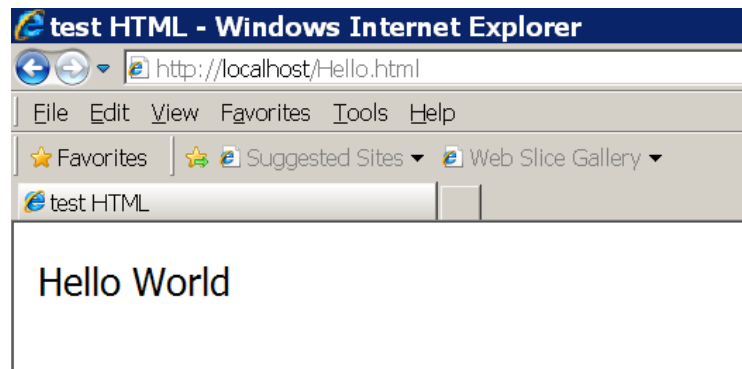
การรันโปรแกรม

การรันโปรแกรมให้ใช้ โปรแกรม Internet Explorer รันโดยให้รันผ่านเครื่อง Server โดยต้องบันทึก ไฟล์ที่จะรันทุกไฟล์ไว้ใน C:\AppServ\www เท่านั้น แล้วใช้รันที่ Browser คำสั่ง <http://localhost/Hello.html> ลองเขียนโปรแกรมตัวอย่าง Hello.html และบันทึกไฟล์ใน C:\AppServ\www แล้วรันโปรแกรมที่แถบ Address Bar

ตัวอย่างที่ 1-1 โปรแกรม Hello World

```
<HTML>
<HEAD>
<TITLE> test HTML </TITLE>
</HEAD>
<BODY>
Hello World
</BODY>
</HTML>
```

ผลลัพธ์



รูปที่ 1-10 : ผลลัพธ์การรันโปรแกรม Hello World

บทที่ 2 การเขียนโปรแกรม HTML

การสร้างเว็บเพจในปัจจุบันใช้ภาษา HTML (เอชทีเอ็มแอล - เป็นคำย่อจากคำเริ่มต้นของ Hypertext Markup Language) เป็นภาษามาร์กอัปหลักในปัจจุบันที่ใช้ในการสร้างเว็บเพจหรือข้อมูลอื่นที่เรียกดูผ่านทางเว็บเบราว์เซอร์ ซึ่งตัวโค้ดจะแสดงโครงสร้างของข้อมูล ในการแสดง ตัวอักษร รูปแบบการแสดงผล ตาราง หัวข้อ ลิงก์ ย่อหน้า รายการ รวมถึงการสร้างแบบฟอร์ม

ภาษา HTML

โครงสร้างของภาษา HTML เริ่มจากแท็ก(tag) HTML : <HTML> หมายถึง การเริ่มต้นเอกสาร HTML และสิ้นสุดด้วยแท็กปิด HTML : </HTML> หมายถึง การสิ้นสุดเอกสาร และมี <HEAD> เพื่อกำหนดรายละเอียดในส่วนหัวของเอกสาร และปิดด้วย </HEAD> ในส่วนหัวเอกสารสามารถแสดงชื่อเรื่อง หรือชื่อเพจของเอกสารได้โดยใช้ แท็กไตเติล <TITLE> และปิดด้วย </TITLE> และในส่วนเนื้อหาของเนื้อหา กำกับด้วย <BODY> ปิดด้วย </BODY> สามารถแสดงโครงสร้างภาษา HTML ได้ดังนี้

```
<HTML>
<HEAD>
<TITLE> </TITLE>
</HEAD>
<BODY>
</BODY>
</HTML>
```

โครงสร้างของภาษา HTML ประกอบด้วย แท็กเปิด <HTML> ซึ่งคู่กันกับ แท็กปิด </HTML> ซึ่งภายใน แท็ก <HTML> ประกอบด้วยแท็ก <HEAD> และ แท็ก <BODY> ในส่วนของ แท็ก HEAD หมายถึง ส่วนหัวของเว็บเพจ และ ในส่วนของ แท็ก Body หมายถึง ส่วนที่เป็น เนื้อหาของเว็บเพจ ข้อความต่าง ๆ ที่ต้องการให้ปรากฏบนหน้าเว็บ จะใส่ในส่วนนี้ ในส่วนของแท็ก Head สามารถเพิ่ม แท็ก Title เข้าไปเพื่อใส่ ชื่อ หรือ ข้อความให้ปรากฏที่ Title bar ของเว็บเพจ

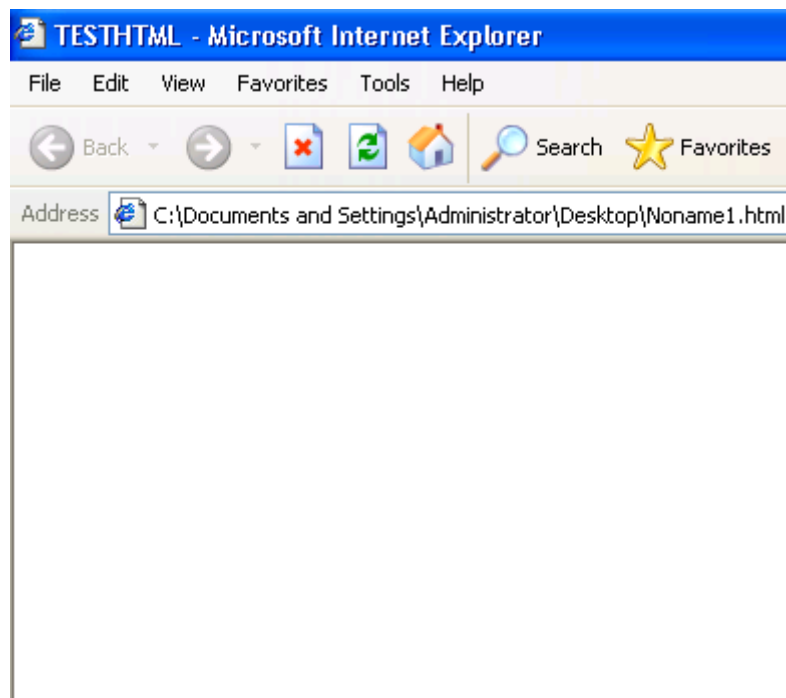
■ คำสั่ง <TITLE></TITLE>

เป็นส่วนแสดงชื่อของเอกสารโดยแสดงที่ไตเติลบาร์ของวินโดว์

ตัวอย่างที่ 2-1 การใส่ชื่อเว็บเพจ

```
<HTML>
<HEAD>
<TITLE> TESTHTML </TITLE>
</HEAD>
<BODY>
</BODY>
</HTML>
```

ผลลัพธ์



รูปที่ 2-1 : ผลลัพธ์การแสดงผลชื่อเว็บเพจ

■ คำสั่ง **<BODY BGCOLOR=Color TEXT=Color> </BODY>**

การแสดงผลเอกสารจะแสดงที่ คำสั่ง BODY เป็นส่วนแสดงข้อความของเอกสาร โดยสามารถกำหนดส่วนขยาย เรียกว่า Attribute เช่น สีพื้นของเว็บเพจ ใช้ = BGCOLOR ชื่อสีที่ต้องการ หรือ รหัสสี # RGB RRGGBB สามารถกำหนดสีตัวอักษรได้โดยกำหนดที่ TEXT = สี หรือ รหัสสี ดังตัวอย่าง

ตัวอย่างที่ 2-2 การใส่สีพื้นและสีตัวอักษร

```
<HTML>
<HEAD>
<TITLE> TEST HTML </TITLE>
</HEAD>
<BODY BGCOLOR=BLUE TEXT=YELLOW>
Welcome
</BODY>
</HTML>
```

ผลลัพธ์



รูปที่ 2-2 : ผลลัพธ์การใส่สีพื้นและสีอักษร

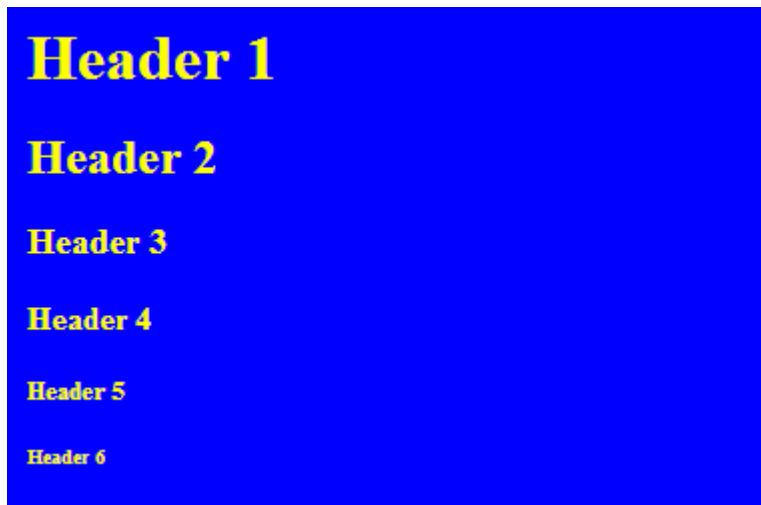
■ คำสั่ง <H#></H#>

คำสั่ง Header เพื่อกำหนดระดับหัวเรื่อง มีตั้งแต่ ระดับที่ 1 ถึง 6 ดังตัวอย่างที่ 3

ตัวอย่างที่ 2-3 การใส่หัวข้อ

```
<BODY BGCOLOR=#0000FF TEXT=#FFFF00>
<H1>Header 1</H1>
<H2>Header 2</H2>
<H3>Header 3</H3>
<H4>Header 4</H4>
<H5>Header 5</H5>
<H6>Header 6</H6>
</BODY>
```

ผลลัพธ์



รูปที่ 2-3 : ผลลัพธ์การใส่หัวข้อ

- คำสั่ง `<FONT FACE=ชื่อแบบอักษร SIZE=ขนาดอักษร COLOR=สีอักษร></FONT>`
คำสั่ง Font เป็นคำสั่งที่ใช้กำหนดรูปแบบตัวอักษร โดยใช้ Attribute FACE กำหนดขนาดให้กับอักษร ใช้ Attribute SIZE และสีของอักษร ด้วย Attribute COLOR

ตัวอย่างที่ 2-4 การกำหนดรูปแบบอักษร

```
<BODY BGCOLOR=#0000FF TEXT=#FFFF00>
<FONT FACE=AngsanaUPC SIZE=5 COLOR=WHITE>WELCOME</FONT>
<FONT FACE="Microsoft Sans Serif" SIZE=6 COLOR=WHITE>WELCOME</FONT>
<FONT FACE="Times New Roman" SIZE=7 COLOR=WHITE>WELCOME</FONT>
</BODY>
```

ผลลัพธ์



รูปที่ 2-4 : ผลลัพธ์การกำหนดหัวข้อ

- คำสั่ง ****
เป็นคำสั่งที่ใช้กำหนดตัวหนาให้กับตัวอักษร
- คำสั่ง **<I></I>**
เป็นคำสั่งที่ใช้กำหนดตัวเอียงให้กับตัวอักษร
- คำสั่ง **<U></U>**
เป็นคำสั่งที่ใช้กำหนดตัวอักษรขีดเส้นใต้
- คำสั่ง ****
เป็นคำสั่งที่ใช้กำหนดตัวอักษรเป็นตัวยก
- คำสั่ง ****
เป็นคำสั่งที่ใช้กำหนดตัวอักษรเป็นตัวห้อย
- การสร้างรายการแบบไม่เรียงลำดับ(Unnumber List)
คำสั่ง **UL** เป็นคำสั่ง กำหนดรายการ แบบไม่เรียงลำดับ โดยสามารถกำหนดลักษณะรายการได้ 3 แบบคือ วงกลมโปร่ง วงกลมทึบและสี่เหลี่ยม ตามตัวอย่าง

```
<UL TYPE=ลักษณะหัวข้อ>
<LI> ข้อความ </LI>
</UL>
```

ตัวอย่างที่ 2-5 การกำหนดรายการ

```
<BODY>
<UL TYPE=square>
<LI>List 1</LI>
<LI>List 2</LI>
</UL>
<UL TYPE=disc>
<LI>List 1</LI>
<LI>List 2</LI>
</UL>
<UL TYPE=disc>
<LI>List 1</LI>
<LI>List 2</LI>
</UL>
</BODY>
```

ผลลัพธ์

- List 1
- List 2

- List 1
- List 2

- List 1
- List 2

■ การสร้างรายการแบบเรียงลำดับ(Ordered List)

คำสั่ง OL เป็นคำสั่ง กำหนดรายการ แบบเรียงลำดับ โดยสามารถกำหนดลักษณะรายการได้ 3 แบบคือ ตัวอักษรพิมพ์เล็ก พิมพ์ใหญ่ เลขโรมัน ตามตัวอย่าง

```
<OL TYPE=ลักษณะหัวข้อ>
<LI> ข้อความ </LI>
</OL>
```

ตัวอย่างที่ 2-6 การกำหนดรายการ

```
<BODY>
<OL TYPE=a>
<LI>List 1</LI>
<LI>List 2</LI>
</OL>

<OL TYPE=I>
<LI>List 1</LI>
<LI>List 2</LI>
</OL>

<OL TYPE=i start=3>
<LI>List 1</LI>
<LI>List 2</LI>
</OL>

</BODY>
```

ผลลัพธ์

- a. List 1
- b. List 2

- I. List 1
- II. List 2

- iii. List 1
- iv. List 2

การสร้างตาราง

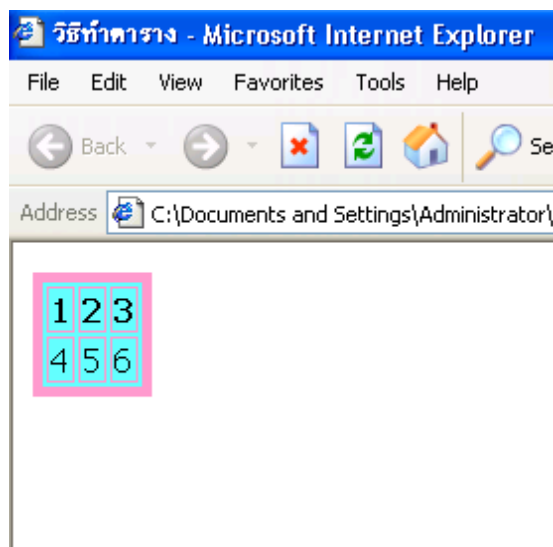
การสร้างตาราง เริ่มจาก แทก Table และปิดด้วย แทก Table ในแทก Table ประกอบด้วย แทก TR หมายถึง แถว ซึ่งย่อมาจาก Table Row ซึ่งต้องเปิดด้วยแทก TR และปิดด้วย แทก TR ในแต่ละแถว หากต้องการมี 5 แถว ก็ต้องมี TR 5 คู่ เท่ากับจำนวนแถวที่ต้องการ และในแต่ละแถวจะมีข้อมูลที่กำหนดด้วยแทก TD กำกับสำหรับข้อมูลในแต่ละตัวด้วย

แทก Table กำหนดรายละเอียดของตารางได้ โดยกำหนดความกว้างของเส้นขอบได้ โดยใช้ Attribute BORDER เช่น Border = 5 หมายถึง ความกว้างของขอบเส้นตาราง =5 กำหนดสีพื้นของตารางได้โดยใช้ Attribute BGCOLOR เช่น bgcolor="#66FFFF" คือกำหนดสีของพื้นตารางเป็นสีฟ้าน้ำทะเล และกำหนดสีเส้นขอบได้โดยใช้ Attribute BORDER COLOR เช่น bordercolor="#FF99CC" คือสีของเส้นตาราง จะเป็นสีชมพูอ่อนดังตัวอย่าง

ตัวอย่างที่ 2-7 การสร้างตาราง

```
<HTML>
<HEAD>
<TITLE> วิธีทำตาราง </TITLE>
</HEAD>
<BODY>
<table border=5 bgcolor="#66FFFF" bordercolor="#FF99CC" >
<TR>
<TH>1</TH><TH>2</TH><TH>3</TH>
</TR>
<TR>
<TD>4</TD><TD>5</TD><TD>6</TD>
</TR>
</table>
</BODY>
</HTML>
```

ผลลัพธ์



รูปที่ 2-5 : ผลลัพธ์การสร้างตาราง

การรวมตาราง มี 2 ลักษณะ คือ

1 การรวมคอลัมน์

การรวมคอลัมน์ คือ การประสานคอลัมน์ เช่น การรวมคอลัมน์ 3 คอลัมน์ เป็น 1 คอลัมน์ หรือ อีกความหมายหนึ่งคือ ขยายความกว้าง เป็น 3 คอลัมน์ ใช้ คำสั่ง `colspan=3`

2 การรวมแถว

การรวมแถว คือการประสานแถว เช่นต้องการประสาน 2 แถว เป็น 1 แถว หรือ ขยายความกว้างของเป็น 2 แถว

■ คำสั่งเพิ่มเติมของตาราง

Cellspacing = 1 คือ ระยะห่างระหว่างตาราง

Cellpadding = 1 คือ ความกว้างภายในกรอบตาราง

ตัวอย่างที่ 2-8 การรวมแถวและคอลัมน์

```
<TABLE BORDER=1 >
<TR><TH colspan=3>1</TH></TR>
<TR><TD rowspan=2 >4</TD><TD >5</TD><TD >6</TD></TR>
<TR>
<TD >8</TD><TD >9</TD></TR>
</TABLE>
```

ผลลัพธ์

1
5 6
4 8 9

ตัวอย่างที่ 2-9 การสร้างตารางและรูปภาพ

```

<HTML><HEAD>
<TITLE>ประวัติทางการศึกษา</TITLE></HEAD>
<BODY text="BLUE">
<center><h2>
ประวัติทางการศึกษา
</h2></center><br>
<center><h2>
<table border=2 bgcolor="LIGHTCYAN" bordercolor="GREEN" >
<TR>
<TH>ปี พ.ศ.</TH><TH>ระดับชั้น</TH><TH>สถานศึกษา</TH>
</TR>
<TR>
<TD align=center>2533</TD><TD align=center>อนุบาล
</TD><TD align=center>โรงเรียนศึกษาวิทยา</TD>
</TR>
<TR>
<TD align=center>2539</TD><TD align=center>ประถมศึกษาปีที่ 6
</TD><TD align=center>โรงเรียนบุรารักษ์</TD>
</TR>
<TR>
<TD align=center>2542</TD><TD align=center>มัธยมศึกษาปีที่ 3
</TD><TD align=center>โรงเรียนบุรารักษ์</TD>
</TR>
<TR>
<TD align=center>2545</TD><TD align=center>มัธยมศึกษาปีที่ 6
</TD><TD align=center>โรงเรียนสมุทรปราการ</TD>
</TR>
<TR>
<TD align=center>2548</TD><TD align=center>กำลังศึกษาปริญญา
ตรีปีที่ 3</TD><TD align=center>มหาวิทยาลัยราชภัฏสวนดุสิต</TD>
</TR>
</table>

<a href="หน้าหลัก.html"> home </a>

</BODY>
</HTML>

```

ผลลัพธ์

ประวัติทางการศึกษา

ปี พ.ศ.	ระดับชั้น	สถานศึกษา
2533	อนุบาล	โรงเรียนศึกษาวิทยา
2539	ประถมศึกษาปีที่ 6	โรงเรียนบุรารักษ์
2542	มัธยมศึกษาปีที่ 3	โรงเรียนบุรารักษ์
2545	มัธยมศึกษาปีที่ 6	โรงเรียนสมุทรปราการ
2548	กำลังศึกษาปริญญาตรีปีที่ 3	มหาวิทยาลัยราชภัฏสวนดุสิต


[home](#)


รูปที่ 2-6 : ผลลัพธ์การสร้างตารางและรูป

- คำสั่ง ****
เป็นคำสั่งที่ใช้แสดงรูปภาพ
- คำสั่ง การจัดรูปแบบอักษร
`align=center/right/left` คือจะเป็นการจัดข้อความหรือภาพในตารางให้อยู่ กลาง/ขวา/ซ้าย
- คำสั่ง การทำ link
`<a href = "ใส่ชื่อ URL ที่จะlink ไป หรือ ชื่อไฟล์ ที่ใช้ link ภายใน"></a>`
- คำสั่ง BR
`<BR>` คือคำสั่งขึ้นบรรทัดใหม่
- คำสั่ง เปิดไฟล์วิดีโอ **<embed src="ชื่อไฟล์ AVI">**
`<Embed src="MOV02021.AVI">` เป็นคำสั่ง เพื่อเปิด ไฟล์ วิดีโอ
- คำสั่ง เปิดเพลง **<bgsound src="ชื่อไฟล์ mp3">**
`<BGSOUND SRC="helloo.wav" LOOP=5>`
`< BGSOUND SRC="helloo.wav" LOOP=INFINITE>`

การสร้างเฟรม Frame

การสร้างเฟรม เป็น การเขียนเว็บแบบโบราณ ซึ่งในปัจจุบันไม่ค่อยนิยมมากนัก เพราะมีโปรแกรมอำนวยความสะดวกในการสร้างเฟรมมากขึ้น การสร้างเฟรมทำให้สามารถแยกส่วนการแสดงผลได้ชัดเจน เช่น ให้แสดงผลในส่วนหนึ่ง ให้แสดงเมนูในส่วนหนึ่ง หรืออีกส่วนหนึ่งอาจจะแสดงโฆษณา โดยจะไม่มี การเปลี่ยนแปลง หากไม่มีการประมวลผลในส่วนนั้น ช่วยในการโหลดข้อมูลได้บางส่วน

การสร้างเฟรม สามารถกำหนดได้ใน 2 รูปแบบ คือ แบ่งเฟรมในแนวแถว หรือ แนวคอลัมน์ โดยใช้คำสั่ง `<frameset rows=" 20% , * ">` เป็นการกำหนดหน้าเพจออกเป็น 2 ส่วน คือ ส่วนบน 20 % และส่วนล่าง คือส่วนที่เหลือ คือ 80 % ดังนั้น ต้องที่ไฟล์ HTML 2 ไฟล์ เพื่อแสดงในส่วนบน และส่วนล่าง

การสร้างเฟรม อีกรูปแบบหนึ่งคือ แบ่งเฟรมในแนวคอลัมน์ โดยใช้คำสั่ง `frameset` แล้วกำหนดเปอร์เซ็นต์ในการแบ่งเฟรม เช่น `<frameset cols=" 20% , * ">` เป็นการกำหนดหน้าเพจออกเป็น 2 ส่วน คือ ส่วนด้านซ้าย 20 % และส่วนด้านขวา คือส่วนที่เหลือ คือ 80 % ดังนั้น ต้องที่ไฟล์ HTML 2 ไฟล์ เพื่อแสดงในส่วนด้านซ้าย และส่วนด้านขวา

การแบ่งเฟรมอาจจะแบ่งในรูปแบบแนวนอน แล้วแบ่งแนวตั้ง อีกครั้ง ก็ได้ ดังตัวอย่างที่ 2.10 โดยเริ่มแรกจะแบ่งเพจในแนวนอน เป็น 2 ส่วน คือส่วนบน 20 % และส่วนที่เหลือ แบ่งอีกในแนวตั้ง โดยแบ่งด้านซ้าย เป็นเมนู 30 % และส่วนที่เหลือ อยู่ด้านขวา เป็นส่วนแสดงผล โดยตั้งชื่อ ส่วนด้านขวา ชื่อ p4 เพื่อใช้ในการอ้างอิงต่อไป

ตัวอย่างที่ 2-10 การสร้างเฟรม

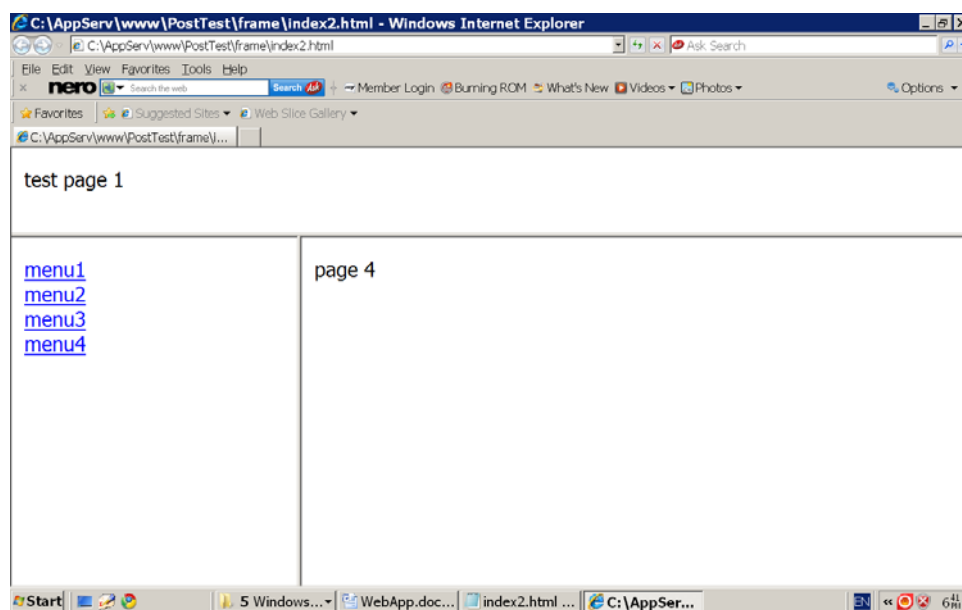
```
<frameset rows="20%,*">  
<frame src="page1.html">  
<frameset cols="30%,*">  
<frame src="page3.html">  
<frame src="page4.html" name="p4">  
</frameset>
```

ในส่วนของเมนู คือไฟล์ Page3.HTML กำหนดลิงค์ โดยแสดงผลในส่วนเฟรมด้านขวา โดยกำหนดใน `Target="p4"` การแสดงผลจะแสดงในส่วนของเฟรมด้านขวา ซึ่งได้กำหนดชื่อไว้แล้ว หากกำหนด `target="blank"` การแสดงผลจะสร้างหน้าต่างใหม่ ให้

ตัวอย่างที่ 2-11 ไฟล์ เมนู Page 3.html

```
<a href="page1.html" target="p4">menu1</a><br>
<a href="page2.html" target="p4">menu2</a><br>
<a href="page3.html" target="blank">menu3</a><br>
<a href="page4.html" target="p4">menu4</a><br>
```

ผลลัพธ์

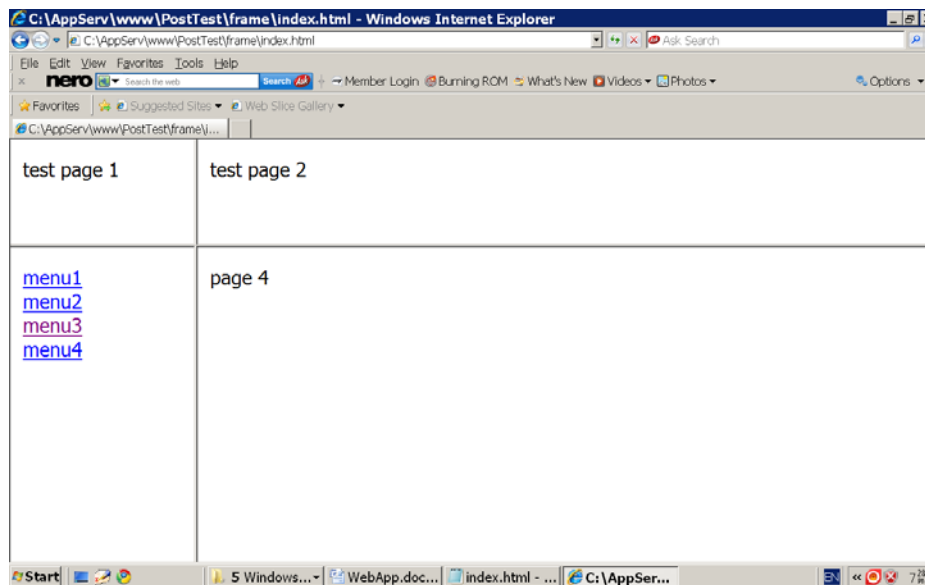


รูปที่ 2-7 : ผลลัพธ์การสร้างเฟรม

ตัวอย่างที่ 2-12 การสร้างเฟรมแบบ 4 เฟรม

```
<FRAMESET ROWS=" 25%, *" COLS="20%, *" frameborder="1"
border="1" framespacing="1">
<FRAME SRC="page1.html" name="p1" >
<FRAME SRC="page2.html" name="p2" >
<FRAME SRC="page3.html" name="p3" >
<FRAME SRC="page4.html" name="p4">
```

ผลลัพธ์



รูปที่ 2-8: ผลลัพธ์การสร้างเฟรม 4 เฟรม

การสร้างฟอร์ม FORM

การสร้างฟอร์ม เป็น การสร้างแบบฟอร์ม เพื่อรับข้อมูลแล้วส่งไปประมวลผลต่อไป ข้อมูลที่จะส่งไปประมวลผลต้องอยู่ภายใน แท็ก FORM

- คำสั่ง **<FORM METHOD=POST ACTION=ชื่อไฟล์></FORM>**

คำสั่ง FORM เป็นคำสั่ง ที่สร้างขอบเขตการส่งข้อมูล โดยมี Attribute METHOD เพื่อกำหนดวิธีการส่งข้อมูล มี Method POST คือ ส่งข้อมูลเมื่อกดปุ่ม Submit และ Attribute ACTION คือ ชื่อไฟล์ ที่จะไปทำงานต่อไป โดยปกติจะเป็นไฟล์ PHP เพื่อรับข้อมูลจากฟอร์ม และทำการคำนวณ หรือ เก็บข้อมูลลงฐานข้อมูล

- คำสั่งที่ใช้ในการสร้าง text box

<input type="text" name="txtName" maxlength=2 size=50>

Attribute type="text" เป็นการกำหนดชนิดของข้อมูลเป็น Text box

Attribute name="txtname" เป็นการกำหนดชื่อ ของ Text box ชื่อ txtname

Attribute maxlength="2" เป็นการกำหนดการรับข้อมูล เป็น 2 อักขร

Attribute size="50" เป็นการกำหนดขนาดความกว้างของ Text box

- คำสั่งที่ใช้ในการสร้าง text area

Text Area คือ การรับข้อมูลแบบอักษร หรือ Text Box แบบหลายบรรทัด

**textArea :: <textarea name = "txtArea" rows=5 cols=50> INPUT
Your Text</textarea>**

- คำสั่งที่ใช้ในการสร้าง List Box

List Box คือ การแสดงข้อมูล แบบหลายบรรทัด

**List Box
<Select Name ="list">
 <option value=usa>America
 <option value=aus>Australia
 <option value=thai selected>Thailand
</Select>**

- คำสั่งที่ใช้ในการสร้าง Radio Button

คำสั่งเพื่อสร้าง Radio Button จะใช้กับข้อมูลกรณีที่ต้องการให้เลือกตัวเลือกได้เพียง 1 ตัวเลือกเท่านั้น

**Radio Button::
เพศ::
<input type="radio" name="gender" value="M">ชาย
<input type="radio" name="gender" value="F">หญิง**

- คำสั่งที่ใช้ในการสร้าง Check box

คำสั่งเพื่อสร้าง Check box จะใช้กับข้อมูลกรณีที่ต้องการให้เลือกตัวเลือกได้มากกว่า 1 ตัวเลือก

<input type="checkbox" name="hobby[]" value="read">อ่านหนังสือ

- คำสั่งที่ใช้ในการสร้างปุ่ม Button

คำสั่งเพื่อใช้ในการสร้างปุ่ม มี 2 แบบ คือ submit เป็นปุ่มที่สามารถส่งข้อมูลจากฟอร์มผ่านไปยังไฟล์ต่อไป และ reset เป็นปุ่มที่สามารถ ยกเลิกการคีย์ข้อมูลที่ได้คีย์ไปแล้ว

**<INPUT TYPE="submit" value = "OK...." name = "OK">
<INPUT TYPE="reset" value="CANCLE" name = "candle">**

■ คำสั่งที่ใช้ในการสร้างปุ่ม Browse

คำสั่ง file เป็นคำสั่งเพื่อรับไฟล์ข้อมูล โดยสามารถกดปุ่ม browse เพื่อเลือกไฟล์ในเครื่อง
ได้

```
<INPUT TYPE="file" name = "file" >
```

ตัวอย่างที่ 2.10 โปรแกรมในการสร้าง Form

```
<FORM>
hello Form<br>
Textbox (^_____)^*<input type="text" name="txtName"
maxlength=2 size=50><br>
Password :: <input type = "password" name="txtPassword"><br>
textArea :: <textarea name = "txtArea" rows=5 cols=50>
INPUT Your Text</textarea><br>
List Box
<Select Name ="list">
  <option value=usa>America
  <option value=aus>Australia
  <option value=thai selected>Thailand
</Select><br><hr>
คุณใช้โทรศัพท์มือถือข่ายใด
<Select Name ="lstMobile[]" multiple size=2>
  <option value=dtac>DTAC
  <option value=ais>AIS
  <option value=orange>Orange
</Select><br>
Radio Button::<br>เพศ::
<input type="radio" name="gender" value="M">ชาย
<input type="radio" name="gender" value="F">หญิง
<br>งานอดิเรกที่ท่านชอบคือ...?
<input type="checkbox" name="hobby[]" value="read">อ่านหนังสือ
<input type="checkbox" name="hobby[]" value="game">เล่นเกม
<input type="checkbox" name="hobby[]" value="music">ฟังเพลง
<input type="checkbox" name="hobby[]" value="sleep">นอน<br>
<INPUT TYPE="hidden" name="id" value="TEST"><br>
<INPUT TYPE="submit" value = "OK...." name = "OK">
<INPUT TYPE="reset" value="CANCLE" name = "candle" >
<INPUT TYPE="file" name = "file" >
</FORM>
```

ผลลัพธ์

hello Form

Textbox (^_____)^*

Password ::

INPUT Your Text

textArea ::

List Box Thailand

คุณใช้โทรศัพท์เครือข่ายใด

Radio Button::

เพศ :: ☐ ชาย ☐ หญิง

งานอดิเรกที่ท่านชอบคือ...? ☐ อ่านหนังสือ ☐ เล่นเกม ☐ ฟังเพลง ☐ นอน

OK.... CANCEL Browse...

■ คำสั่งที่ใช้ในการข้อมูลไฟล์

คำสั่ง file เป็นคำสั่งเพื่อรับไฟล์ข้อมูล โดยสามารถกดปุ่ม browse เพื่อเลือกไฟล์ในเครื่องได้

<INPUT TYPE="file" name = "file" >

คำสั่งเพื่อใช้ในการรับข้อมูลไฟล์ มักจะใช้กับการรับข้อมูลที่เป็นไฟล์ เช่น ไฟล์รูปภาพ ในการระบุว่าพนักคนนี้มีรูปร่างหน้าตา อย่างไร หรือรูปของสินค้า หรือต้องการ upload ข้อมูลที่เป็นไฟล์ เป็นต้น

การส่งข้อมูลไฟล์ มีผลเกี่ยวข้องกับการสร้างฟอร์ม เพื่อการรับและส่งข้อมูลด้วย ดังนั้น เมื่อต้องการส่งข้อมูลแบบไฟล์ ต้องกำหนด Attribute Enctype = Multipart เพิ่มเติม มิฉะนั้น ข้อมูลไฟล์ดังกล่าว จะไม่สามารถ POST ไปตามต้องการ

```
<FORM METHOD=POST ACTION="Tst.php" ENCTYPE = "multipart/form-data">
  Picture :: <INPUT TYPE="file" name = "txtPic"><BR>
  <INPUT TYPE="submit" value = "OK...." name = "OK">
  <INPUT TYPE="reset" value="CANCLE" name = "cancle" >
</FORM>
```

ผลลัพธ์

Picture ::

พระราชดำรัสเศรษฐกิจพอเพียง

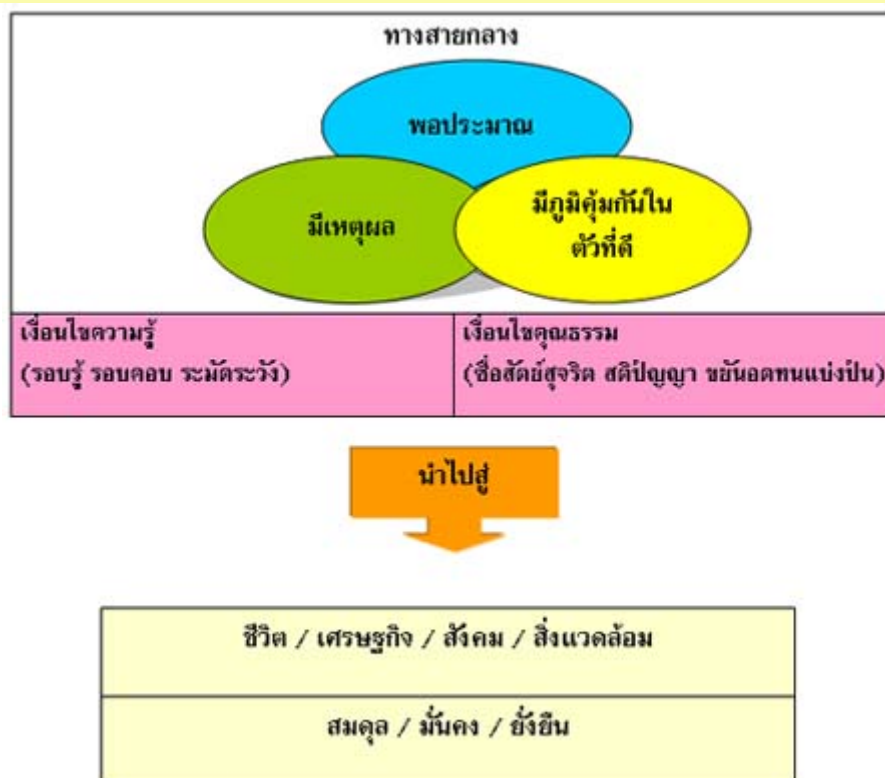
เศรษฐกิจพอเพียงเป็นเสมือนรากฐานของชีวิต รากฐานความมั่นคงของแผ่นดิน เปรียบเสมือนเสาเข็มที่ถูกตอกรองรับบ้านเรือนตัวอาคารไว้นั่นเอง สิ่งก่อสร้างจะมั่นคงได้ก็อยู่ที่เสาเข็ม แต่คนส่วนมากมองไม่เห็นเสาเข็มและลืมเสาเข็มเสียด้วยซ้ำไป

พระราชดำรัสพระบาทสมเด็จพระเจ้าอยู่หัว จากวารสารชัยพัฒนา

“เศรษฐกิจพอเพียง... จะทำความเจริญให้แก่ประเทศได้ แต่ต้องมีความเพียร แล้วต้องอดทน ต้องไม่ใจร้อน ต้องไม่พูดมาก ต้องไม่ทะเลาะกัน ถ้าทำโดยเข้าใจกัน เชื่อว่าทุกคนจะมีความพอใจได้...”

พระราชดำรัสพระบาทสมเด็จพระเจ้าอยู่หัว พระราชทาน ณ วันที่ ๔ ธันวาคม ๒๕๔๑

หลักปรัชญาเศรษฐกิจพอเพียง ของ พระบาทสมเด็จพระเจ้าอยู่หัว



บทที่ 3 ภาษาจาวาสคริปต์

ภาษาจาวาสคริปต์ (JavaScript Language) คือ ภาษาสคริปต์ประเภทไคลเอนต์สคริปต์ ซึ่งจะทำงานบนเครื่องไคลเอนต์ เป็นภาษาสคริปต์แบบหนึ่งที่มีโครงสร้างคล้ายภาษาซี มีคำสั่งการทำงานแบบเงื่อนไข การวนซ้ำ การสร้างฟังก์ชัน และมีการเขียนแบบออบเจกต์ และการเขียนตามเหตุการณ์ โดยสามารถควบคุมการทำงาน และสามารถตรวจสอบข้อมูลได้ก่อนที่จะส่งข้อมูลให้กับเครื่องเซิร์ฟเวอร์

รูปแบบการเขียนโปรแกรม Java Script

การเขียนโปรแกรมภาษาจาวาสคริปต์ เป็นภาษาสคริปต์ ที่ใช้ในการเขียนเว็บแบบไดนามิกเว็บ เรียกว่า DHTML (Dynamic Hyper Text Markup Language) รูปแบบของภาษาจึงมีลักษณะคล้ายกับภาษา HTML ฟังก์ชันการทำงานเขียนคล้ายภาษาซี แต่ละคำสั่งคั่นด้วยเครื่องหมายเซมิโคลอน ; การเขียนจาวาสคริปต์ต้องใช้ความระมัดระวังอย่างสูงเพราะเป็นภาษาสคริปต์ ไม่มีคอมไพเลอร์แปลภาษาให้ก่อน ดังนั้นจึงไม่มีการตรวจสอบรหัสคำสั่งให้ก่อนเมื่อเขียนคำสั่งผิด โปรแกรมจะไม่ทำงานและไม่มีข้อผิดพลาดให้เห็น ดังนั้นต้องพิมพ์คำสั่งอย่างตั้งใจอย่างมาก ตัวเล็กตัวใหญ่ มีผลต่อการรันผลลัพธ์ โดยเฉพาะฟังก์ชันการทำงาน

- รูปแบบการเขียนแบบภาษา JavaScript

```
<SCRIPT LANGUAGE="JavaScript">  
คำสั่งของภาษา JavaScript ;  
</SCRIPT>
```

- คำสั่งการแสดงผล document.write

```
<SCRIPT LANGUAGE="JavaScript">  
document.write(" Hello World ");  
</SCRIPT>
```

ผลลัพธ์

Hello World

- คำสั่งการแสดงผลแบบ alert

```
<SCRIPT LANGUAGE="JavaScript">
    alert (" Hello World " );
</SCRIPT>
```

ผลลัพธ์



การเขียนหมายเหตุในโปรแกรมภาษา Java Script

การเขียนหมายเหตุ หรือ comment เพื่ออธิบายความหมาย หรือเพื่อกำหนดไม่ให้เกิดการทำงานของโปรแกรม ของภาษา JavaScript มีหลายรูปแบบด้วยกัน คือ

- รูปแบบการเขียนหมายเหตุแบบภาษาซี แบบ 1 บรรทัด

```
<SCRIPT LANGUAGE="JavaScript">
    document.write("comment 1 line") ;
    // comment
</SCRIPT>
```

- รูปแบบการเขียนหมายเหตุแบบภาษาซี แบบหลายบรรทัด

```
<SCRIPT LANGUAGE="JavaScript">
    document.write("comment multi line") ;
    /* comment line 1
       comment line 2 */
</SCRIPT>
```

การประกาศตัวแปรของโปรแกรมภาษา Java Script

การประกาศตัวแปรของภาษา JavaScript สามารถกำหนดตัวแปร โดยกำหนดค่าให้กับตัวแปร โดยไม่ต้องระบุว่าเป็นชนิดใด โดยโปรแกรมจะกำหนดให้อัตโนมัติ โดยดูจากข้อมูลที่กำหนดให้

- รูปแบบการประกาศตัวแปร

```
var ชื่อตัวแปร ;
```

- ตัวอย่างการประกาศตัวแปร

```
var name ;  
var I ;  
var flag ;
```

- รูปแบบการประกาศตัวแปร และกำหนดค่าเริ่มต้น

```
var ชื่อตัวแปร = ค่าเริ่มต้น ;
```

- ตัวอย่างการประกาศตัวแปร และกำหนดค่าเริ่มต้น

```
var name = "Napatsarun" ;  
var I = 1 ;  
var flag = true ;
```

- รูปแบบกำหนดค่าเริ่มต้นให้กับตัวแปร

```
ชื่อตัวแปร = ค่าเริ่มต้น ;
```

- ตัวอย่างกำหนดค่าเริ่มต้นให้กับตัวแปร

```
name = "Napatsarun" ;  
I = 2 ;  
flag = true ;
```

ตัวแปรของภาษาจาวาสคริปต์ สามารถนำไปใช้ได้โดยไม่ต้องประกาศตัวแปรก่อน โดยโปรแกรมจะกำหนดให้ ตามข้อมูลที่กำหนดให้เริ่มต้น เช่น ตัวแปร `name = "Napatsarun"` ตัวแปร `name` จะเป็นชนิดข้อมูลข้อความ `String` ตัวแปร `i` จะเป็นชนิดข้อมูลตัวเลข `Number` และตัวแปร `flag` จะเป็นชนิดข้อมูลตรรกะ `Boolean` ซึ่งมีค่าเป็นจริง (`true`) หรือเท็จ (`false`)

- รูปแบบการสร้างตัวแปรอาร์เรย์

ชื่อตัวแปร = new array (จำนวนสมาชิก);

- ตัวอย่างการสร้างตัวแปรอาร์เรย์

month = new array (3) ;

- รูปแบบกำหนดค่าให้กับตัวแปรอาร์เรย์

ชื่อตัวแปร [index] = ข้อมูล ;

- ตัวอย่างการกำหนดค่าให้กับตัวแปรอาร์เรย์

```
month = new array (3) ;
month [0] = "Jan" ;
month [1] = "Feb" ;
month [2] = "Mar" ;
```

การรับค่าและการแสดงผลตัวแปร

ตัวแปรนอกจากกำหนดให้ด้วยการกำหนดค่าเริ่มต้นให้แล้ว บางกรณีอาจจำเป็นต้องรับค่าจากผู้ใช้

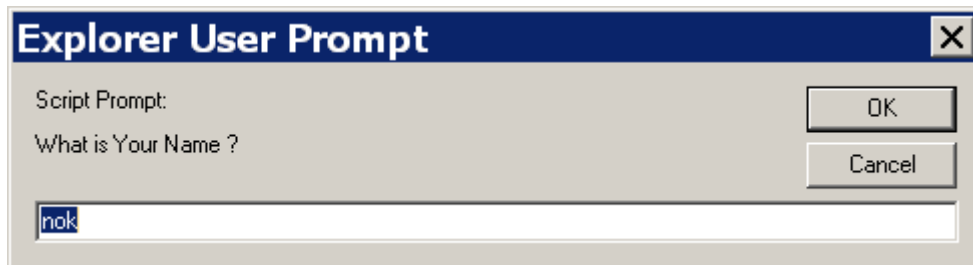
- การรับค่าด้วยคำสั่ง `prompt`

ชื่อตัวแปร = prompt ("ข้อความ") ;

- ตัวอย่างการรับค่าด้วยคำสั่ง prompt

```
name = prompt ("What is your name ? " , "Nok") ;
```

ผลลัพธ์



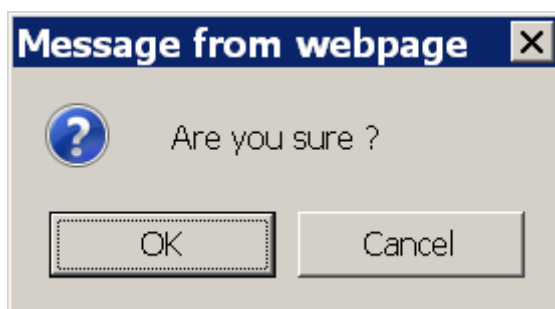
- การยืนยันตอบรับ ด้วยคำสั่ง confirm

```
ชื่อตัวแปร = confirm ("ข้อความ" ) ;
```

- ตัวอย่างการยืนยันตอบรับ ด้วยคำสั่ง confirm

```
ans=confirm("Are you sure ? ");  
document.write(" Answer = " + ans );
```

ผลลัพธ์



กรณีเลือก OK

Answer = true

กรณีเลือก Cancel

Answer = false

การแปลงชนิดข้อมูล

ตัวแปรที่ได้จากการรับค่า บางกรณีต้องแปลงเป็นชนิดข้อมูลตัวเลขเพื่อทำการคำนวณ หรือข้อมูลที่ได้จากการคำนวณ อาจจำเป็นต้องแสดงผลในรูปแบบต่าง ๆ การแปลงชนิดข้อมูล จึงจำเป็นต้องใช้ประกอบในการเขียนโปรแกรม

- การแปลงตัวเลขเป็นตัวอักษร ด้วย method `.toString()`

ตัวแปร`Number.toString()` ;

- ตัวอย่างการแปลงตัวเลขเป็นตัวอักษร ด้วย method `.toString()`

```
var str = "23" ;  
var num = 0 ;  
str=num.toString();
```

- การแปลงตัวอักษรเป็นตัวเลข ด้วยคำสั่ง `parseInt()`

parseInt(String) ;

- ตัวอย่างการแปลงตัวอักษรเป็นตัวเลข ด้วยคำสั่ง `parseInt()`

```
var str = "23" ;  
var num = 0 ;  
num = parseInt(str);
```

- การแปลงตัวอักษรเป็นตัวเลขทศนิยม ด้วยคำสั่ง `parseFloat()`

parseFloat(String) ;

- ตัวอย่างการแปลงตัวอักษรเป็นตัวเลขทศนิยม ด้วยคำสั่ง `parseFloat()`

```
var str = "23.5" ;  
var num = 0.0 ;  
num = parseFloat(str);
```

ตัวดำเนินการ

ตัวดำเนินการที่ใช้ในการคำนวณคณิตศาสตร์ เหมือนกับภาษาซี คือ บวก(+) ลบ(-) คูณ(*)หาร(/) และ หารเอาเศษ (%) หรือว่า mod ดังแสดงความหมายในตาราง

ตารางที่ 3-1 ตัวดำเนินการทางคณิตศาสตร์

ตัวดำเนินการ	ตัวอย่าง	ความหมาย
บวก (+)	$a + b$	หาผลบวกของตัวแปร $a + b$
	$9 + 5$	หาผลบวกของค่า $9 + 5$ มีค่าเท่ากับ 14
ลบ (-)	$a - b$	หาผลลัพธ์ของค่าในตัวแปร $a - b$
	$9 - 5$	หาผลลัพธ์ของค่า $9 - 5$ มีค่าเท่ากับ 4
คูณ (*)	$a * b$	หาผลลัพธ์ของค่าตัวแปร $a * b$
	$9 * 5$	หาผลลัพธ์ของค่า $9 * 5$ มีค่าเท่ากับ 45
หาร (/)	a / b	หาผลลัพธ์ของตัวแปร a / b
	$9 / 5$	หาผลลัพธ์ของค่า $9 / 5$ มีค่าเท่ากับ 45
หารเอาเศษ (%)	$a \% b$	หาเศษของการหาร a ด้วย b เหลือเศษ 4
	$9 \% 5$	หาเศษของการหาร 9 ด้วย 5 เหลือเศษ 4

ตารางที่ 3-2 ตัวดำเนินการทางคณิตศาสตร์แบบลดรูป

ตัวดำเนินการ	ตัวอย่าง	ความหมาย
+=	$a += b$	$a=a+b$; สมมติ $a = 10$; $b = 20$; จะได้ $a = 30$
	$a += 5$	$a=a+5$; สมมติ $a = 10 + 5$; จะได้ $a = 15$
-=	$a -= b$	$a=a-b$; สมมติ $a = 10$; $b = 2$; จะได้ $a = 8$
	$a -= 5$	$a=a-5$; สมมติ $a = 10 - 5$; จะได้ $a = 5$
*=	$a *= b$	$a=a*b$; สมมติ $a = 10$; $b = 2$; จะได้ $a = 20$
	$a *= 5$	$a=a*5$; สมมติ $a = 10 * 5$; จะได้ $a = 50$
/=	$a /= b$	$a=a/b$; สมมติ $a = 10$; $b = 5$; จะได้ $a = 2$
	$a /= 5$	$a=a/5$; สมมติ $a = 10 / 5$; จะได้ $a = 2$
%=	$a \% = b$	$a=a\%b$; สมมติ $a = 10$; $b = 5$; จะได้ $a = 0$
	$a \% = 5$	$a=a\%5$; สมมติ $a = 10 \% 5$; จะได้ $a = 0$

ตารางที่ 3-3 แสดงการเพิ่มค่า และ การลดค่า

ตัวดำเนินการ	ตัวอย่าง	ความหมาย
เพิ่มค่า (++)	b=a++	กำหนดค่าก่อนแล้วเพิ่มค่า สมมติ ค่า a = 9 จะได้ค่า b = 9
	b=++a	กำหนดค่าหลังเพิ่มค่า สมมติ ค่า a = 9 จะได้ค่า b = 10
ลดค่า (--)	b=a--	กำหนดค่าก่อนลดค่า สมมติ ค่า a = 9 จะได้ค่า b = 9
	b=--a	กำหนดค่าหลังลดค่า สมมติ ค่า a = 9 จะได้ค่า b = 8

ตารางที่ 3-4 แสดงตัวดำเนินการเปรียบเทียบ

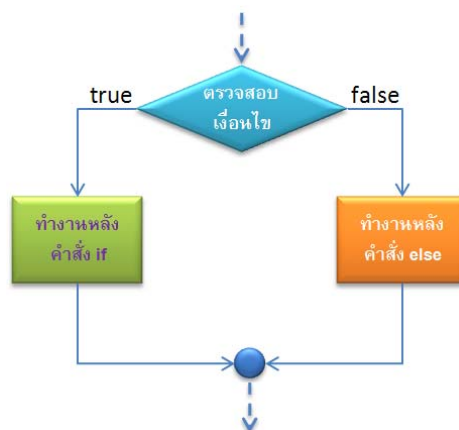
ตัวเปรียบเทียบ	ความหมาย	ตัวอย่าง
==	เท่ากับ	\$A == \$B\$ เปรียบเทียบว่า A เท่ากับ B หรือไม่
!=	ไม่เท่ากับ	\$A != \$B\$ เปรียบเทียบว่า A ไม่เท่ากับ B หรือไม่
>	มากกว่า	\$A > \$B\$ เปรียบเทียบว่า A มากกว่า B หรือไม่
>=	มากกว่าหรือเท่ากับ	\$A >= \$B\$ เปรียบเทียบว่า A มากกว่าหรือเท่ากับ B หรือไม่
<	น้อยกว่า	\$A < \$B\$ เปรียบเทียบว่า A น้อยกว่า B หรือไม่
<=	น้อยกว่าหรือเท่ากับ	\$A <= \$B\$ เปรียบเทียบว่า A น้อยกว่าหรือเท่ากับ B หรือไม่
&&	และ	\$A > 4 && \$B > 5\$ เปรียบเทียบว่า A > 4 และ B > 5 หรือไม่
	หรือ	\$A > 4 \$B > 5\$ เปรียบเทียบว่า A > 4 หรือ B > 5 หรือไม่

การเขียนโปรแกรมแบบมีเงื่อนไข

การเขียนโปรแกรมแบบมีเงื่อนไข ต้องเกิดขึ้นเสมอในกรณี ที่ มีการตัดสินใจเพื่อทำอย่างใดอย่างหนึ่ง หรือ ไม่กระทำการอย่างหนึ่ง หรือมีเงื่อนไขเพื่อให้โปรแกรมทำตามเงื่อนไขที่ผู้ใช้ต้องการ ซึ่งจะนิยมใช้คำสั่ง if เพราะสามารถสร้างเงื่อนไขการทำงาน ได้ครอบคลุมทุกกรณี เช่น มากกว่า น้อยกว่า เท่ากัน ไม่เท่ากัน ได้ครบทุกกรณี และสามารถเช็ค และ หรือ ได้ อีกด้วย กรณีที่มีมากกว่า 1 เงื่อนไข ซึ่งสามารถแยกเป็นประเภทได้ดังนี้

เงื่อนไข 1 เงื่อนไข มีทางเลือก 2 ทาง

การเขียนโปรแกรมแบบมี 1 เงื่อนไข มีทางเลือก 2 ทางเลือก จะใช้คำสั่ง if-else ในการตัดสินใจเมื่อเงื่อนไขเป็นจริง (true) โปรแกรมจะเลือกทำงานคำสั่งหลัง if แต่หากเงื่อนไขเป็นเท็จ โปรแกรมจะเลือกทำงานคำสั่งหลัง else ซึ่งคำสั่งหลัง if หรือคำสั่ง หลัง else อาจเป็นคำสั่งเดียว หรือเป็นชุดคำสั่งก็ได้ ถ้าเป็นชุดคำสั่ง ก็ใส่ปีกกา ครอบชุดคำสั่งไว้ และที่สำคัญ เงื่อนไขต้องอยู่ภายในวงเล็บ



รูปที่ 3-1 : ผังการไหลคำสั่ง if แบบ 2 ทางเลือก

if (เงื่อนไข)

 คำสั่ง

else

 คำสั่ง

กรณีที่เป็นชุดคำสั่ง

if (เงื่อนไข)

{

<ชุดคำสั่ง>

}

else

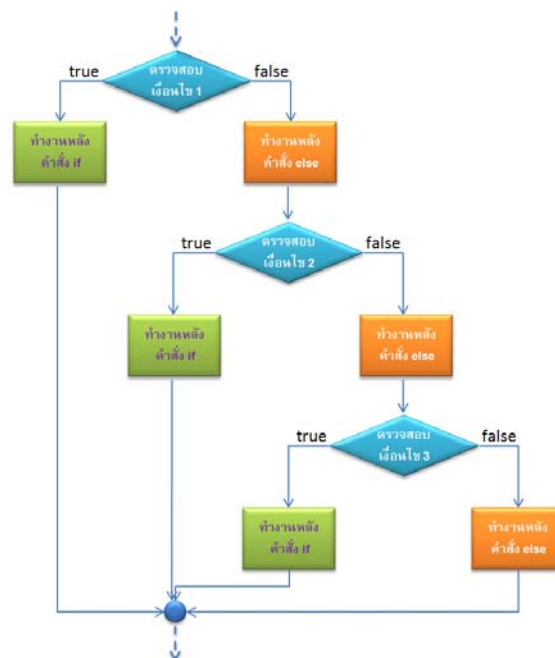
{

<ชุดคำสั่ง>

}

เงื่อนไขมากกว่า 1 เงื่อนไข

การเขียนโปรแกรมบางครั้ง เงื่อนไขมีมากกว่า 1 เงื่อนไข จึงต้องมีคำสั่ง if-elseif เพื่อตัดสินใจเลือก จากทางเงื่อนไขมากกว่า 1 เงื่อนไข เช่น การตัดเกรดนักศึกษา เป็นกรณีที่เห็นได้ชัดเจนว่า ต้องมีเงื่อนไขในการกำหนดเกรด แต่ละเกรด ซึ่งไม่สามารถใช้ เงื่อนไขเดียวเพื่อตัดเกรดนักศึกษา 5 เกรดได้



รูปที่ 3-2 : ผังการไหลคำสั่ง if แบบหลายทางเลือก

```

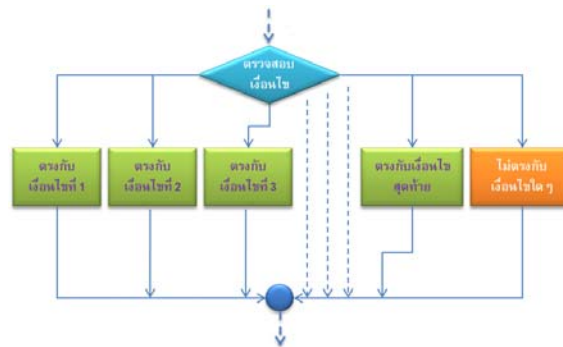
if (เงื่อนไขที่ 1)
{
    <ชุดคำสั่งเมื่อเงื่อนไขที่ 1 ตรวจสอบแล้วเป็น true>
}
else if (เงื่อนไขที่ 2)
{
    <ชุดคำสั่งเมื่อเงื่อนไขที่ 2 ตรวจสอบแล้วเป็น true>
}
else if (เงื่อนไขที่ 3)
{
    <ชุดคำสั่งเมื่อเงื่อนไขที่ 3 ตรวจสอบแล้วเป็น true>
}
else if
    :
    :
else
{
    <ชุดคำสั่งเมื่อไม่ตรงกับเงื่อนไขใดๆ เลย>
}

```

การทำงานของคำสั่ง if-elseif นั้น โปรแกรมจะทำงานตรวจสอบเงื่อนไขที่ 1 ก่อน หากเงื่อนไขที่ 1 เป็นจริง จะทำงานตามคำสั่งหลัง if หากเงื่อนไขที่ 1 ไม่เป็นจริง จะตรวจสอบเงื่อนไขที่ 2 ต่อไป หากเงื่อนไขที่ 2 เป็นจริง โปรแกรมจะทำงานตามคำสั่ง หลังเงื่อนไขที่ 2 หากเงื่อนไขที่ 2 ไม่เป็นจริง จะทำการตรวจสอบเงื่อนไข ต่อไป หากไม่มีเงื่อนไขไหนเป็นจริงเลย จะทำงานตามคำสั่งหลัง else

ตัดสินใจเลือกมากกว่า 2 ทางเลือก โดยใช้ switch-case

คำสั่ง switch case ที่มีความยืดหยุ่นพอสมควร สามารถ ใช้ได้ในหลายกรณี คือ สามารถเปรียบเทียบ ค่าเดียว หลายค่า ค่าตัวแปร ตัวเลข และ ข้อความ รวมถึง ค่าตัวแปร Boolean



รูปที่ 3-3 : ผังการไหลคำสั่ง switch – case

ในการใช้งานนั้นมีรูปแบบดังนี้

```

switch (เงื่อนไข)
{
    case เงื่อนไขแรก :
        คำสั่ง ;
        break ;
    case เงื่อนไขที่สอง :
        คำสั่ง ;
        break ;
    case สุดท้าย :
        คำสั่ง ;
        break ;
    default:
        <เมื่อไม่ตรงกับเงื่อนไขใดๆ เลย>
}
  
```

ตัวอย่างที่ 3-1 การใช้ switch case

```

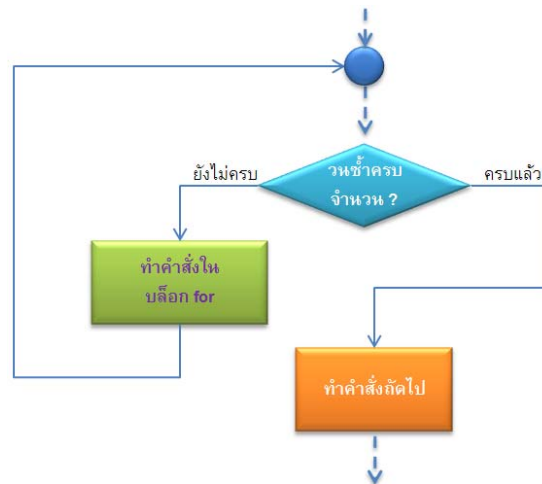
switch (totaltime) {
    case (totaltime < 1):
        document.write( "That was fast!" ) ;
        break;

    case (totaltime > 1):
        document.write( "Not fast!" ) ;
        break;

    case (totaltime > 10):
        document.write("That's sloooooow" ) ;
        break;
}
  
```

การเขียนโปรแกรมแบบวนซ้ำ

การเขียนโปรแกรมแบบวนซ้ำ มีหลายคำสั่ง หากรู้จำนวนรอบที่แน่นอน หรือ มี index ในการแสดงผล ส่วนใหญ่จะใช้ for ในการวนลูป



รูปที่ 3-4 : ผังการไหลคำสั่ง for

ในการใช้งานคำสั่ง for นั้นมีรูปแบบดังนี้

```

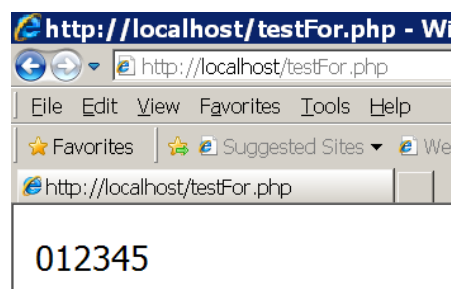
for (ตัวแปรใช้นับ = ค่าเริ่มต้น ; เงื่อนไขการหยุดวนซ้ำ ; Step ขั้นของการนับ)
{
    < ทำงานตามคำสั่ง >
}
  
```

คำสั่ง for ประกอบด้วย 3 ส่วน คือ ส่วนแรก เป็นการกำหนดค่าเริ่มต้น ให้กับตัวแปรที่ใช้เป็น index เช่น $i=1$ ส่วนที่สอง คือ เงื่อนไขการจบการทำงาน อาจจะมากกว่า น้อยกว่า ค่าใดค่าหนึ่ง หรือตัวแปร ก็ได้ เช่น $i \leq 5$ ส่วนสุดท้าย คือ การเพิ่มค่า หรือ การลดค่า การเพิ่มค่าขึ้นหนึ่งจะใช้ $i++$ ลดค่าลงหนึ่ง ก็ใช้ $i--$ โดยสามารถกำหนดการเพิ่มหรือลดได้ หรืออาจจะใช้เพิ่มค่า ครั้งละ 2 ก็ได้ โดยใช้ $i+=2$ หากต้องการเพิ่มครั้งละ 2 หรือ $i=i+2$ ก็ได้ $i+=2$ มีผลเท่ากับ $i=i+2$ ซึ่ง $i+=2$ เป็นการเขียนแบบลัดรูป

ตัวอย่างที่ 3-2 การใช้ for

```
for(i=0; i<=5; i++)
{
    document.write (i) ;
}
```

ผลลัพธ์

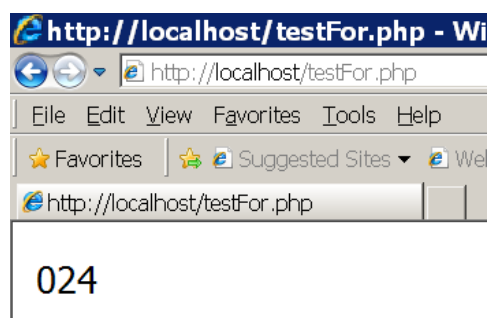


รูปที่ 3-5 : ผลลัพธ์ การใช้คำสั่ง for

ตัวอย่างที่ 3-3 การใช้ for โดยเพิ่มค่า ครั้งละ 2

```
for(i=0; i<=5; i+=2)
{
    document.write( i ) ;
}
```

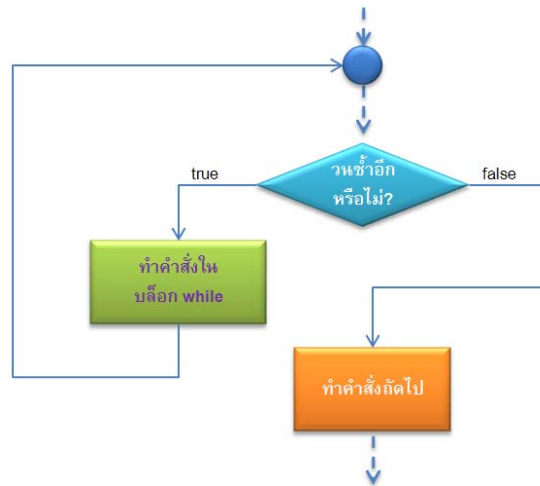
ผลลัพธ์



รูปที่ 3-6 : ผลลัพธ์ การใช้คำสั่ง for โดยเพิ่มครั้งละ 2

การวนซ้ำด้วยการใช้ while

การวนลูปด้วยการใช้ while เป็นคำสั่งที่มีการตรวจสอบเงื่อนไขก่อน แล้วจึงจะทำงาน



รูปที่ 3-7 : ผังการไหลคำสั่ง while

ในการใช้งาน while นั้นมีรูปแบบดังนี้

```

while (เงื่อนไข)
{
    < ชุดคำสั่ง >
}
  
```

ตัวอย่างที่ 3-4 การใช้ while

```

var i = 1;
while (i <= 10) {
    document.write (i++);
}
  
```

การวนซ้ำด้วยการใช้ do-while

ปกติการใช้ while นั้นเราจะตรวจสอบเงื่อนไขก่อนการวนซ้ำในแต่ละครั้ง แต่ถ้าเราต้องการให้การตรวจสอบเงื่อนไขไปอยู่ด้านหลัง (คือต้องวนซ้ำอย่างน้อย 1 รอบก่อน) เราจะใช้ do มาช่วย โดยจะนำเงื่อนไขการตรวจสอบวนซ้ำ while ไปไว้ด้านหลัง ดังรูปแบบต่อไปนี้ ยังมีรูปแบบการวนซ้ำในลักษณะวนซ้ำด้วยจำนวนรอบที่ไม่แน่นอนแล้ว อีกหลายรูปแบบ ได้แก่

```
do
{
    < ทำงานตามคำสั่ง >
}
while (เงื่อนไข) ;
```

ตัวอย่างที่ 3-5 การใช้ do while

```
i = 0;
do {
    document.write( i ) ;
} while (i > 0);
```

การสร้างฟังก์ชัน

การสร้างฟังก์ชัน มีรูปแบบคล้ายกับฟังก์ชันในภาษาซี คือ มีการกำหนดชื่อฟังก์ชัน พารามิเตอร์ และมีคำสั่งการทำงานในฟังก์ชัน

```
function ชื่อฟังก์ชัน (พารามิเตอร์ 1 , พารามิเตอร์ 2 , ... ) {

    คำสั่งการทำงาน ;
    return (ตัวแปร) ;
}
```

```
function Show (x ) {
    document.write (x) ;
    return (x) ;
}
```

การเรียกใช้งานฟังก์ชัน

ชื่อฟังก์ชัน (ตัวแปร) ;

ตัวอย่างที่ 3-6 การเรียกใช้งานฟังก์ชัน

```
<script>
  function show()
  {
    document.write("Hello" );
  }
</script>

<a href="javascript:show()">show</a>
<input type="button" value="Show" onClick="show()">
```

ผลลัพธ์

[show](#)

เมื่อคลิกที่ show จะแสดงข้อความ Hello

Hello

- ตัวอย่างการเรียกใช้งานฟังก์ชันส่งค่า

```

<script>
  function show(x)
  {
    document.write(x);
  }
</script>

<script>
a=prompt("input value :",0);
</script>
<a href="javascript:show(a)">show</a>
<input type="button" value="Show" onClick="show(a)">

```

ผลลัพธ์



ตัวอย่างที่ 3-7 โปรแกรมแสดงค่าจากการเลือก List Box

```

<form method=post>
<input type=text name=docfile>
<select name=file onclick=javascript:docfile.value=file.value;>
<option value="">Please Select if you have a file
<option value='3271-bb.jpg'>3271-bb.jpg
<option value='bank.mdb'>bank.mdb
<option value='tftpd32.gif'>tftpd32.gif
</select>
</form>

```

ผลลัพธ์

3271-bb.jpg

3271-bb.jpg

คนเราเกิดมาแล้ว ไม่ยอมเรียน หรือจะรู้ ไม่ยอมดู หรือจะเห็น ไม่ยอมฟัง หรือจะได้ยิน
ไม่ยอมทำ หรือจะเป็น จะลำเค็ญอย่าแหยจนแก่ตาย

เกษตรตามแนว "ทฤษฎีใหม่"

"ทฤษฎีใหม่" เป็นแนวทางหรือหลักในการบริหารจัดการที่ดินและน้ำ เพื่อการเกษตรในที่ดินขนาดเล็กให้เกิดประโยชน์สูงสุด ด้วยหลักเศรษฐกิจพอเพียง พระบาทสมเด็จพระเจ้าอยู่หัวได้พระราชทานพระราชดำรินี้เพื่อเป็นการช่วยเหลือเกษตรกรที่ประสบความยากลำบาก ให้สามารถผ่านช่วงวิกฤต โดยเฉพาะการขาดแคลนน้ำได้โดยไม่เดือดร้อนและยากลำบากนัก

การดำเนินงานตามทฤษฎีใหม่มี 3 ขั้นตอน คือ

- 1) การผลิต ให้พึ่งตนเองด้วยวิธีง่าย ค่อยเป็นค่อยไปตามกำลัง ให้พอมีพอกิน
- 2) การรวมพลังกันในรูปแบบ หรือ สหกรณ์ ร่วมแรงร่วมใจกัน ในด้านการผลิต การตลาด ความเป็นอยู่ สวัสดิการ การศึกษา สังคมและศาสนา
- 3) การดำเนินธุรกิจโดยติดต่อ ประสานงาน จัดหาทุนหรือแหล่งเงิน

ในขั้นแรกที่เป็นการผลิต ถือเป็นขั้นสำคัญที่สุด ให้แบ่งออกเป็น 4 ส่วน ตามอัตราส่วน 30 : 30 : 30 : 10 หมายถึง

ขุดสระเก็บกักน้ำ

พื้นที่ประมาณ 30% ให้ขุดสระเก็บกักน้ำ เพื่อให้มีน้ำใช้สม่ำเสมอตลอดปี โดยเก็บกักน้ำฝนในฤดูฝน และใช้เสริมการปลูกพืชในฤดูแล้ง หรือระยะฝนทิ้งช่วง ตลอดจนการเลี้ยงสัตว์ และพืชน้ำต่างๆ เช่น ผักบุ้ง ผักกระเฉด โสน ฯลฯ

ปลูกข้าว

พื้นที่ประมาณ 30 % ให้ปลูกข้าวในฤดูฝน เพื่อใช้เป็นอาหารประจำวันสำหรับครัวเรือนให้เพียงพอตลอดปี โดยไม่ต้องซื้อหาในราคาแพง เป็นการลดค่าใช้จ่าย และสามารถพึ่งตนเองได้

ปลูกผลไม้ ไม้ยืนต้น พืชไร่ พืชผัก

พื้นที่ประมาณ 30 % ให้ปลูกไม้ผล ไม้ยืนต้น พืชไร่ พืชผัก พืชสมุนไพร ฯลฯ อย่างผสมผสานกัน และหลากหลายในพื้นที่เดียวกัน เพื่อใช้เป็นอาหารประจำวัน หากเหลือจากการบริโภคก็นำไปขายได้



เป็นที่อยู่อาศัย และอื่น ๆ

พื้นที่ประมาณ 10 % ใช้เป็นที่อยู่อาศัย เลี้ยงสัตว์ ถนนหนทาง คันดิน โรงเรือนและสิ่งก่อสร้างอื่น ๆ รวมทั้ง คอกเลี้ยงสัตว์ เรือนเพาะชำ นางเก็บผลผลิตผลการเกษตร ฯลฯ

- หลักการและแนวทางสำคัญในการดำเนินงานเกษตรตามแนว "ทฤษฎีใหม่" ที่ควรทราบมีดังนี้
 - เป็นระบบการผลิตแบบเศรษฐกิจพอเพียง ที่เกษตรกรสามารถเลี้ยงตัวเองได้ในระดับที่ประหยัดก่อน
 - ต้องมีพื้นที่ส่วนหนึ่งทำนาข้าว เพราะข้าวเป็นปัจจัยหลักที่ทุกครัวเรือนต้องปลูก เพื่อให้มีข้าวพอบริโภคตลอดทั้งปี
 - ต้องมีน้ำสำรองไว้ใช้เพียงพอตลอดปี เพื่อการเพาะปลูกในระยะฝนทิ้งช่วง หรือในฤดูแล้ง
 - ใช้อัตราส่วน 30 : 30 : 30 : 10 ในการแบ่งพื้นที่ออกเป็น 4 ส่วน ไม่ว่าจะมีความถี่ที่ถ่อรองน้อยกว่าหรือมากกว่า 15 ไร่ คือ
 - 30 % ใช้ขุดสระเก็บกักน้ำ
 - 30 % ใช้ปลูกข้าว
 - 30 % ใช้ปลูกพืชผัก ผลไม้ พืชไร่ ไม้ยืนต้น
 - 10 % ใช้เป็นที่อยู่อาศัยและอื่นๆ
- ประโยชน์ของ "ทฤษฎีใหม่" สรุปได้ดังนี้
 - ประชาชนพออยู่พอกินในระดับประหยัด เลี้ยงตนเองได้ ไม่อดอยาก ตามหลักปรัชญาของ "เศรษฐกิจพอเพียง"
 - ในหน้าแล้งก็สามารถนำน้ำที่เก็บกักไว้ในสระมาปลูกพืชที่ใช้น้ำน้อย เช่น ถั่วต่างๆ ได้ โดยไม่ต้องอาศัยชลประทาน
 - ในปีที่ฝนตกตามฤดูกาล ก็สามารถสร้างรายได้ให้ร่ำรวยได้
 - ในกรณีที่เกิดอุทกภัย ก็สามารถฟื้นตัว และช่วยตัวเองได้ในระดับหนึ่ง โดยราชการไม่ต้องช่วยเหลือมากนัก

ในการจัดการเกษตรตามแนว "ทฤษฎีใหม่" ให้เกิดประสิทธิภาพนั้น ประเด็นสำคัญ คือ การพึ่งตนเอง ประหยัด และมีธรรมาภิบาล ก่อให้เกิดเศรษฐกิจพอเพียง โดยการจัดการแบ่งพื้นที่ให้สัมพันธ์ และเกื้อกูลกัน ซึ่งเป็นการบริหารจัดการทรัพยากรที่ดิน น้ำ แรงงาน และรักษาสภาพแวดล้อมให้ดีขึ้น อันจะนำไปสู่การผลิตที่เกิดรายได้ และสู่การพัฒนาที่ยั่งยืน.

ที่มา http://www.baanjommyut.com/library/new_land/

บทที่ 4 การเขียนโปรแกรม PHP

การเขียนโปรแกรม PHP มีลักษณะรูปแบบคล้ายการเขียนโปรแกรมซี การเขียนโปรแกรม PHP ใช้สำหรับเขียนโปรแกรมเว็บดาต้าเบส คือสามารถเขียนโปรแกรมควบคุมการทำงานบนเว็บ และสามารถดึงข้อมูลจากฐานข้อมูลมาแสดงในรูปแบบเว็บเพจได้ นอกจากนี้ PHP มีการเขียนโปรแกรมสามารถแสดงข้อมูล มีตัวดำเนินการคำนวณ การเขียนโปรแกรมตามเงื่อนไข การเขียนโปรแกรมควบคุมการทำงาน การทำซ้ำ

การรันโปรแกรม PHP ต้องรันผ่าน Server และโปรแกรมที่เขียนต้องบันทึกไว้ที่ C:\AppServ\www ซึ่งเป็น Path ที่กำหนดโดย AppServ และการรันโปรแกรมต้องรันผ่าน Server ใช้ http://localhost หรือ http://127.0.0.1 ก็ได้

รูปแบบการเขียนโปรแกรม PHP

รูปแบบการเขียนโปรแกรม PHP ซึ่งเป็นโปรแกรมที่รันบน Browser รูปแบบการเขียนเป็นสคริปต์ และการรันต้องรันผ่าน Server ซึ่งมีรูปแบบการเขียนอยู่หลายแบบด้วยกัน ที่นิยมเขียนกันมี 3 แบบ คือ

- รูปแบบการเขียนแบบ SGML

```
<?
    คำสั่งของภาษา PHP
?>
```

- รูปแบบการเขียนแบบภาษา XML

```
<?php
    คำสั่งของภาษา PHP
?>
```

- รูปแบบการเขียนในลักษณะการเขียนสคริปต์ JavaScript

```
<SCRIPT LANGUAGE="PHP">
    คำสั่งของภาษา PHP
</SCRIPT>
```

การเขียนหมายเหตุในโปรแกรมภาษา PHP

การเขียนหมายเหตุ หรือ comment เพื่ออธิบายความหมาย หรือเพื่อกำหนดไม่ให้เกิดการทำงานของโปรแกรม ของภาษา PHP มีหลายรูปแบบด้วยกัน คือ

- รูปแบบการเขียนหมายเหตุแบบ perl

```
<?
    คำสั่งของภาษา PHP
    # comment
?>
```

- รูปแบบการเขียนหมายเหตุแบบภาษาซี แบบ 1 บรรทัด

```
<?
    คำสั่งของภาษา PHP
    // comment
?>
```

- รูปแบบการเขียนหมายเหตุแบบภาษาซี แบบหลายบรรทัด

```
<?
    คำสั่งของภาษา PHP
    /* comment line 1
       comment line 2 */
?>
```

คำสั่งพื้นฐานของโปรแกรมภาษา PHP

คำสั่งของภาษา PHP มีหลายลักษณะการทำงานเหมือนกับการเขียนโปรแกรมภาษาทั่วไป โดยลักษณะส่วนใหญ่จะคล้ายคลึงกับภาษาซีมาก เช่น ตัวดำเนินการ การสร้างเงื่อนไข การวนซ้ำ

คำสั่งแสดงผล

คำสั่งแสดงผล มี 2 คำสั่งที่ใช้กันบ่อย ๆ คือ คำสั่ง echo และคำสั่ง print ซึ่งทั้ง 2 คำสั่งเป็นคำสั่งที่ใช้ในการแสดงผล ใช้แทนกันได้ โดยมีรูปแบบดังนี้

■ คำสั่ง echo

คำสั่ง echo ใช้แสดงข้อความ และตัวแปร

```
echo "ข้อความ"
echo $ตัวแปร
```

■ คำสั่ง print

คำสั่ง print ใช้แสดงข้อความ และตัวแปร

```
print "ข้อความ"
print $ตัวแปร
```

ตัวอย่างที่ 4-1 การแสดงผล

```
1  <?
2  $a=123;
3  echo "Hello";
4  echo $a;
5  ?>
```

ผลลัพธ์

Hello123

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 <? เป็นการเปิดคำสั่ง PHP

บรรทัดที่ 2 \$a=123 เป็นการกำหนดค่าให้กับตัวแปร a โดยตัวแปรต้องมีเครื่องหมาย \$ นำหน้าเสมอ

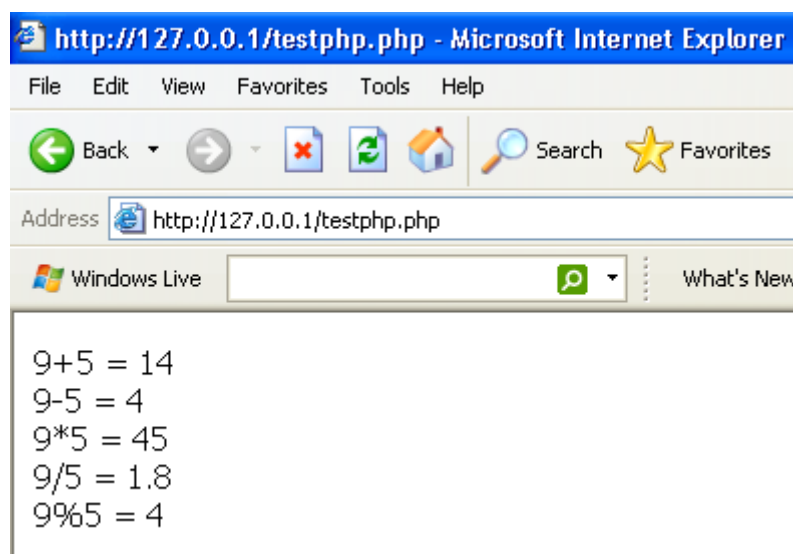
บรรทัดที่ 3 echo "Hello" เป็นการแสดงคำว่า Hello ซึ่งสามารถใช้คำสั่ง print แทน echo ได้

บรรทัดที่ 4 echo \$a เป็นการแสดงค่าในตัวแปร a ออกมา ซึ่งคือ 123

บรรทัดที่ 5 ?> เป็นการปิดคำสั่ง PHP

หมายเหตุ

โปรแกรมที่เขียนต้องบันทึก หรือ save เป็นนามสกุล .php เช่น testphp.php และต้องเก็บไว้ที่ C:\AppServ\www เท่านั้น หรือ path ของ server ถ้าเป็น IIS จะใช้ C:\inetpub\wwwroot เป็นต้น และเวลารันโปรแกรม ต้องรันที่ Browser Internet Explorer หรือ Browser ตัวอื่นด้วยคำสั่ง `http://localhost/ชื่อไฟล์.php` หรือ `http://127.0.0.1/ชื่อไฟล์.php` เช่น `http://localhost/testphp.php` หรือ `http://127.0.0.1/testphp.php` ดังแสดงในรูปผลลัพธ์



รูปที่ 4-1 : ผลลัพธ์การแสดงผล

ตัวดำเนินการทางคณิตศาสตร์

ตัวดำเนินการที่ใช้ในการคำนวณคณิตศาสตร์ เหมือนกับภาษาซี คือ บวก(+) ลบ(-) คูณ(*)หาร(/) และ หารเอาเศษ (%) หรือว่า mod ดังแสดงความหมายในตาราง

ตารางที่ 4-1 ตัวดำเนินการทางคณิตศาสตร์

ตัวดำเนินการ	ตัวอย่าง	ความหมาย
บวก (+)	$\$a + \b	หาผลบวกของตัวแปร $a + b$
	$9 + 5$	หาผลบวกของค่า $9 + 5$ มีค่าเท่ากับ 14
ลบ (-)	$\$a - \b	หาผลลัพธ์ของค่าในตัวแปร $a - b$
	$9 - 5$	หาผลลัพธ์ของค่า $9 - 5$ มีค่าเท่ากับ 4
คูณ (*)	$\$a * \b	หาผลลัพธ์ของค่าตัวแปร $a * b$
	$9 * 5$	หาผลลัพธ์ของค่า $9 * 5$ มีค่าเท่ากับ 45
หาร (/)	$\$a / \b	หาผลลัพธ์ของตัวแปร a / b
	$9 / 5$	หาผลลัพธ์ของค่า $9 / 5$ มีค่าเท่ากับ 45
หารเอาเศษ (%)	$\$a \% \b	หาเศษของการหาร a ด้วย b เหลือเศษ 4
	$9 \% 5$	หาเศษของการหาร 9 ด้วย 5 เหลือเศษ 4

ตารางที่ 4-2 ตัวดำเนินการทางคณิตศาสตร์แบบลดรูป

ตัวดำเนินการ	ตัวอย่าง	ความหมาย
+=	$\$a += \b	$a=a+b$; สมมติ $a = 10$; $b = 20$; จะได้ $a = 30$
	$\$a += 5$	$a=a+5$; สมมติ $a = 10 + 5$; จะได้ $a = 15$
-=	$\$a -= \b	$a=a-b$; สมมติ $a = 10$; $b = 2$; จะได้ $a = 8$
	$\$a -= 5$	$a=a-5$; สมมติ $a = 10 - 5$; จะได้ $a = 5$
*=	$\$a *= \b	$a=a*b$; สมมติ $a = 10$; $b = 2$; จะได้ $a = 20$
	$\$a *= 5$	$a=a*5$; สมมติ $a = 10 * 5$; จะได้ $a = 50$
/=	$\$a /= \b	$a=a/b$; สมมติ $a = 10$; $b = 5$; จะได้ $a = 2$
	$\$a /= 5$	$a=a/5$; สมมติ $a = 10 / 5$; จะได้ $a = 2$
%=	$\$a \% = \b	$a=a\%b$; สมมติ $a = 10$; $b = 5$; จะได้ $a = 0$
	$\$a \% = 5$	$a=a\%5$; สมมติ $a = 10 \% 5$; จะได้ $a = 0$

ตัวอย่างที่ 4-2 ตัวดำเนินการ

```

1  <?
2  $a=9;
3  $b=5;
4  print " $a+$b = ".$a+$b."<br>";
5  print " $a-$b = ".$a-$b."<br>";
6  print " $a*$b = ".$a*$b."<br>";
7  print " $a/$b = ".$a/$b."<br>";
8  print " $a%$b = ".$a%$b."<br>";
9  ?>

```

ผลลัพธ์

$$9+5 = 14$$

$$9-5 = 4$$

$$9*5 = 45$$

$$9/5 = 1.8$$

$$9\%5 = 4$$

อธิบายรหัสคำสั่ง

บรรทัดที่ 2 และ 3 เป็นการกำหนดค่าให้กับตัวแปร a และ ตัวแปร b ให้ โดยกำหนดให้ตัวแปร a = 9 และ ตัวแปร b = 5

บรรทัดที่ 4 ถึง 8 เป็นคำสั่ง เพื่อแสดงผลการทำงานของตัวดำเนินการต่าง ๆ การเชื่อมต่อระหว่าง ข้อความ กับตัวแปร ใช้เครื่องหมายจุด (.) ในการต่อประโยค และใส่วงเล็บเพื่อคำนวณหาผลลัพธ์ ก่อน แสดงผล

ตัวดำเนินการเพิ่มค่าและลดค่า

ตัวดำเนินการที่ใช้ในการเพิ่มค่าและลดค่า เหมือนกับภาษาซี คือ การเพิ่มค่า(++) การลดค่า(--) ซึ่งมีความหมายสำหรับเครื่องหมาย ที่อยู่ข้างหน้า หรือข้างหลังตัวแปร มีความหมายต่างกัน เช่น การเพิ่มค่า โดยมีเครื่องหมายเพิ่มค่าอยู่ข้างหน้า จะมีการเพิ่มค่าก่อน แล้วจึงกำหนดค่าให้กับตัวแปร หากเครื่องหมายเพิ่มค่าอยู่ข้างหลัง จะกำหนดค่าให้กับตัวแปรก่อน แล้วจึงเพิ่มค่าตัวเองขึ้นหนึ่ง ซึ่งการใช้คำสั่งกำหนดค่าแบบนี้ ต้องใช้ความระมัดระวังเป็นพิเศษ หากไม่แน่ใจ ให้ใช้การแยกเป็นสองคำสั่งคือ คำสั่งเพิ่มค่า และคำสั่ง การกำหนดค่า แยกออกจากกัน ส่วนการลดค่าก็เช่นเดียวกัน ดังแสดงความหมายในตารางที่ 4-3

ตารางที่ 4-3 แสดงการเพิ่มค่า และ การลดค่า

ตัวดำเนินการ	ตัวอย่าง	ความหมาย
เพิ่มค่า (++)	\$b=\$a++	กำหนดค่าก่อนแล้วเพิ่มค่า สมมติ ค่า a = 9 จะได้ค่า b = 9
	\$b=++\$a	กำหนดค่าหลังเพิ่มค่า สมมติ ค่า a = 9 จะได้ค่า b = 10
ลดค่า (--)	\$b=\$a--	กำหนดค่าก่อนลดค่า สมมติ ค่า a = 9 จะได้ค่า b = 9
	\$b=--\$a	กำหนดค่าหลังลดค่า สมมติ ค่า a = 9 จะได้ค่า b = 8

ตัวอย่างที่ 4-3 ตัวดำเนินการลดค่า

```

1  <?
2  $a=9;
3  $c=$a--;
4  $d=--$a;
5  print "c=".$c."<br>";
6  print "d=".$d;
7  ?>

```

ผลลัพธ์

c=9

d=7

อธิบายรหัสคำสั่ง

บรรทัดที่ 2 เป็นการกำหนดค่าให้กับตัวแปร a ให้ a = 9

บรรทัดที่ 3 เป็นคำสั่ง เพื่อกำหนดค่า ในตัวแปร a ซึ่งมีค่า เท่ากับ 9 ให้กับตัวแปร c เท่ากับ 9 ด้วย แล้วจึงลดค่าตัวเองลง 1 เหลือ 8 จากบรรทัดที่ 3 ค่า a มีค่าเท่ากับ 8 แล้ว แต่ค่า c ยังคงเท่ากับ 9

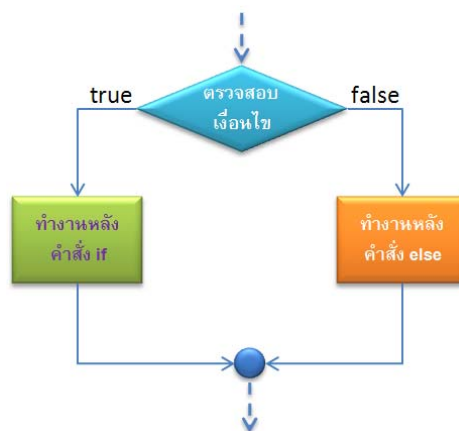
บรรทัดที่ 4 เป็นคำสั่ง เพื่อกำหนดค่าลดค่า ในตัวแปร a ซึ่งมีค่า เท่ากับ 8 แล้ว ลดลง อีก 1 เหลือ 7 แล้วกำหนดค่าให้กับตัวแปร d เท่ากับ 7 ด้วย จากบรรทัดที่ 4 ค่า a มีค่าเท่ากับ 7 และค่า d เท่ากับ 7 ด้วย

การเขียนโปรแกรมแบบมีเงื่อนไข

การเขียนโปรแกรมแบบมีเงื่อนไข ต้องเกิดขึ้นเสมอในกรณี ที่ มีการตัดสินใจเพื่อทำอย่างใดอย่างหนึ่ง หรือ ไม่กระทำอย่างหนึ่ง หรือมีเงื่อนไขเพื่อให้โปรแกรมทำตามเงื่อนไขที่ผู้ใช้ต้องการ ซึ่งจะนิยมใช้คำสั่ง if เพราะสามารถสร้างเงื่อนไขการทำงาน ได้ครอบคลุมทุกกรณี เช่น มากกว่า น้อยกว่า เท่ากัน ไม่เท่ากัน ได้ครบทุกกรณี และสามารถเช็ค และ หรือ ได้ อีกด้วย กรณีที่มีมากกว่า 1 เงื่อนไข ซึ่งสามารถแยกเป็นประเภทได้ดังนี้

เงื่อนไข 1 เงื่อนไข มีทางเลือก 2 ทาง

การเขียนโปรแกรมแบบมี 1 เงื่อนไข มีทางเลือก 2 ทางเลือก จะใช้คำสั่ง if-else ในการตัดสินใจเมื่อเงื่อนไขเป็นจริง (true) โปรแกรมจะเลือกทำงานคำสั่งหลัง if แต่หากเงื่อนไขเป็นเท็จ โปรแกรมจะเลือกทำงานคำสั่งหลัง else ซึ่งคำสั่งหลัง if หรือคำสั่ง หลัง else อาจเป็นคำสั่งเดียว หรือเป็นชุดคำสั่งก็ได้ ถ้าเป็นชุดคำสั่ง ก็ใส่ปีกกา ครอบชุดคำสั่งไว้ และที่สำคัญ เงื่อนไขต้องอยู่ภายในวงเล็บ



รูปที่ 4-2 : ผังการไหลคำสั่ง if แบบ 2 ทางเลือก

if (เงื่อนไข)

คำสั่ง

else

คำสั่ง

กรณีที่เป็นชุดคำสั่ง

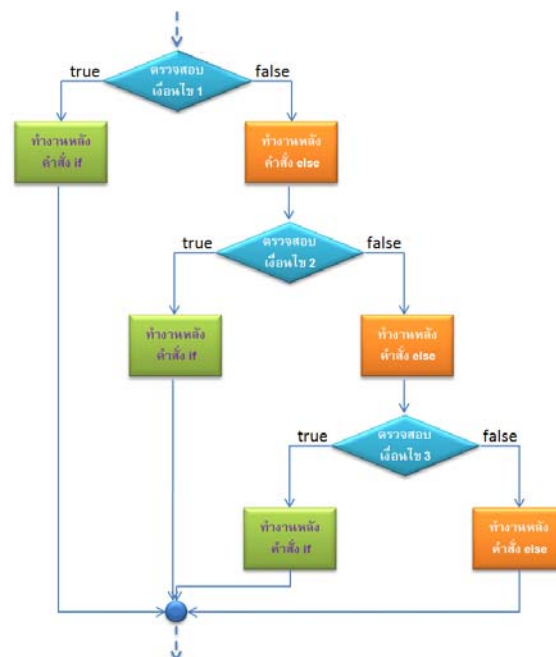
```

if (เงื่อนไข)
{
    <ชุดคำสั่ง>
}
else
{
    <ชุดคำสั่ง>
}

```

เงื่อนไขมากกว่า 1 เงื่อนไข

การเขียนโปรแกรมบางครั้ง เงื่อนไขมีมากกว่า 1 เงื่อนไข จึงต้องมีคำสั่ง if-elseif เพื่อตัดสินใจเลือก จากทางเงื่อนไขมากกว่า 1 เงื่อนไข เช่น การตัดเกรดนักศึกษา เป็นกรณีที่เห็นได้ชัดเจนว่า ต้องมีเงื่อนไขในการกำหนดเกรด แต่ละเกรด ซึ่งไม่สามารถใช้ เงื่อนไขเดียวเพื่อตัดเกรดนักศึกษา 5 เกรดได้



รูปที่ 4-3 : ผังการไหลคำสั่ง if แบบหลายทางเลือก

```

if (เงื่อนไขที่ 1)
{
    <ชุดคำสั่งเมื่อเงื่อนไขที่ 1 ตรวจสอบแล้วเป็น true>
}
else if (เงื่อนไขที่ 2)
{
    <ชุดคำสั่งเมื่อเงื่อนไขที่ 2 ตรวจสอบแล้วเป็น true>
}
else if (เงื่อนไขที่ 3)
{
    <ชุดคำสั่งเมื่อเงื่อนไขที่ 3 ตรวจสอบแล้วเป็น true>
}
else if
:
:
else
{
    <ชุดคำสั่งเมื่อไม่ตรงกับเงื่อนไขใดๆ เลย>
}

```

การทำงานของคำสั่ง if-elseif นั้น โปรแกรมจะทำงานตรวจสอบเงื่อนไขที่ 1 ก่อน หากเงื่อนไขที่ 1 เป็นจริง จะทำงานตามคำสั่งหลัง if หากเงื่อนไขที่ 1 ไม่เป็นจริง จะตรวจสอบเงื่อนไขที่ 2 ต่อไป หากเงื่อนไขที่ 2 เป็นจริง โปรแกรมจะทำงานตามคำสั่ง หลังเงื่อนไขที่ 2 หากเงื่อนไขที่ 2 ไม่เป็นจริง จะทำการตรวจสอบเงื่อนไข ต่อไป หากไม่มีเงื่อนไขไหนเป็นจริงเลย จะทำงานตามคำสั่งหลัง else

ตัวดำเนินการเปรียบเทียบ

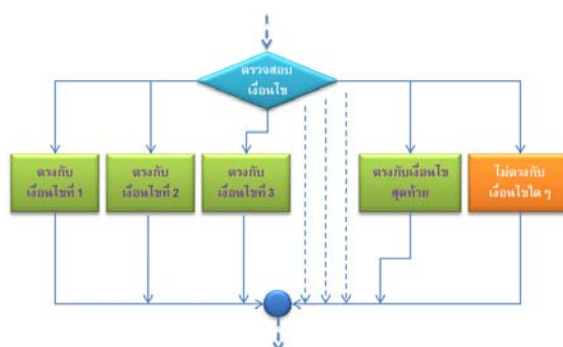
การสร้างเงื่อนไข ในบางกรณี ต้องการตรวจสอบเงื่อนไขมากกว่า 1 อย่างในการทำงานแต่ละครั้ง จึงต้องตัวดำเนินการเปรียบเทียบเข้ามาช่วยในกรณี ที่ต้องการตรวจสอบเงื่อนไขมากกว่า 1 อย่าง

ตารางที่ 4-4 แสดงตัวดำเนินการเปรียบเทียบ

ตัว เปรียบเทียบ	ความหมาย	ตัวอย่าง
==	เท่ากับ	<code>\$A == \$B</code> เปรียบเทียบว่า A เท่ากับ B หรือไม่
!=	ไม่เท่ากับ	<code>\$A != \$B</code> เปรียบเทียบว่า A ไม่เท่ากับ B หรือไม่
>	มากกว่า	<code>\$A > \$B</code> เปรียบเทียบว่า A มากกว่า B หรือไม่
>=	มากกว่าหรือ เท่ากับ	<code>\$A >= \$B</code> เปรียบเทียบว่า A มากกว่าหรือเท่ากับ B หรือไม่
<	น้อยกว่า	<code>\$A < \$B</code> เปรียบเทียบว่า A น้อยกว่า B หรือไม่
<=	น้อยกว่าหรือ เท่ากับ	<code>\$A <= \$B</code> เปรียบเทียบว่า A น้อยกว่าหรือเท่ากับ B หรือไม่
&&	และ	<code>\$A > 4 && \$B > 5</code> เปรียบเทียบว่า <code>\$A > 4</code> และ <code>\$B > 5</code> หรือไม่
	หรือ	<code>\$A > 4 \$B > 5</code> เปรียบเทียบว่า <code>\$A > 4</code> หรือ <code>\$B > 5</code> หรือไม่

switch-case : ตัดสินใจเลือกมากกว่า 2 ทางเลือก

PHP มีคำสั่ง switch case ที่มีความยืดหยุ่นพอสมควร สามารถใช้ได้หลายกรณี คือ สามารถเปรียบเทียบ ค่าเดียว หลายค่า ค่าตัวแปร ตัวเลข และ ข้อความ รวมถึง ค่าตัวแปร Boolean



รูปที่ 4-4 : ผังการไหลคำสั่ง switch – case

ในการใช้งานนั้นมีรูปแบบดังนี้

```
switch (เงื่อนไข)
{
    case เงื่อนไขแรก :
        คำสั่ง ;
        break ;
    case เงื่อนไขที่สอง :
        คำสั่ง ;
        break ;
    case สุดท้าย :
        คำสั่ง ;
        break ;
    default:
        <เมื่อไม่ตรงกับเงื่อนไขใดๆ เลย>
}
```

ตัวอย่างที่ 4-4 การใช้ switch case

```
<?php
switch ($totaltime) {

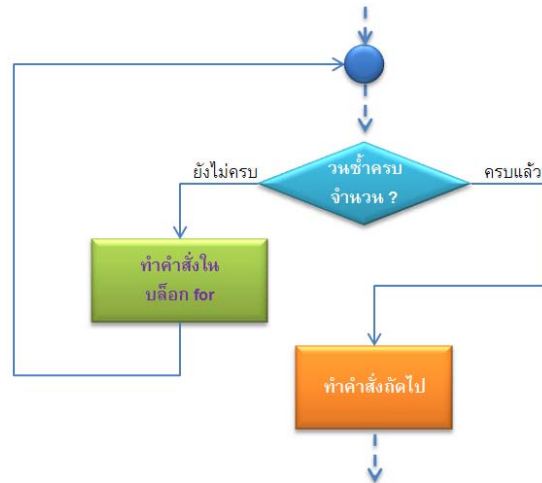
    case ($totaltime < 1):
        echo "That was fast!";
        break;

    case ($totaltime > 1):
        echo "Not fast!";
        break;

    case ($totaltime > 10):
        echo "That's sloooooow";
        break;
}
?>
```

การเขียนโปรแกรมแบบวนซ้ำ

การเขียนโปรแกรมแบบวนซ้ำ มีหลายคำสั่ง หากรู้จำนวนรอบที่แน่นอน หรือ มี index ในการแสดงผล ส่วนใหญ่จะใช้ for ในการวนลูป



รูปที่ 4-5 : ผังการไหลคำสั่ง for

ในการใช้งานคำสั่ง for นั้นมีรูปแบบดังนี้

```

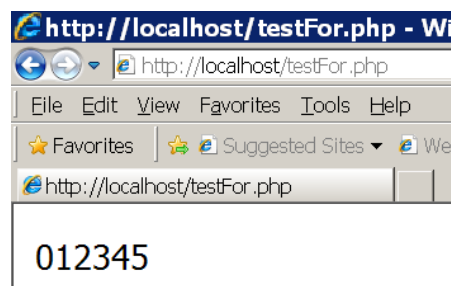
for (ตัวแปรใช้นับ = ค่าเริ่มต้น; เงื่อนไขการหยุดวนซ้ำ; Step ขั้นของการนับ)
{
    < ทำงานตามคำสั่ง >
}
  
```

คำสั่ง for ประกอบด้วย 3 ส่วน คือ ส่วนแรก เป็นการกำหนดค่าเริ่มต้น ให้กับตัวแปรที่ใช้เป็น index เช่น $i=1$ ส่วนที่สอง คือ เงื่อนไขการจบการทำงาน อาจจะมากกว่า น้อยกว่า ค่าใดค่าหนึ่ง หรือตัวแปร ก็ได้ เช่น $i \leq 5$ ส่วนสุดท้าย คือ การเพิ่มค่า หรือ การลดค่า การเพิ่มค่าขึ้นหนึ่งจะใช้ $i++$ ลดค่าลงหนึ่ง ก็ใช้ $i--$ โดยสามารถกำหนดการเพิ่มหรือลดได้ หรืออาจจะใช้เพิ่มค่า ครั้งละ 2 ก็ได้ โดยใช้ $i+=2$ หากต้องการเพิ่มครั้งละ 2 หรือ $i=i+2$ ก็ได้ $i+=2$ มีผลเท่ากับ $i=i+2$ ซึ่ง $i+=2$ เป็นการเขียนแบบลัดรูป

ตัวอย่างที่ 4-5 การใช้ for

```
<?php
for($i=0; $i<=5; $i++)
{
    echo $i ;
}
?>
```

ผลลัพธ์

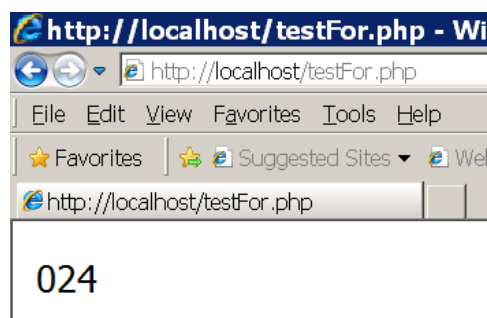


รูปที่ 4-6 : ผลลัพธ์ การใช้คำสั่ง for

ตัวอย่างที่ 4-6 การใช้ for โดยเพิ่มค่า ครั้งละ 2

```
<?php
for($i=0;$i<=5;$i+=2)
{
    echo $i ;
}
?>
```

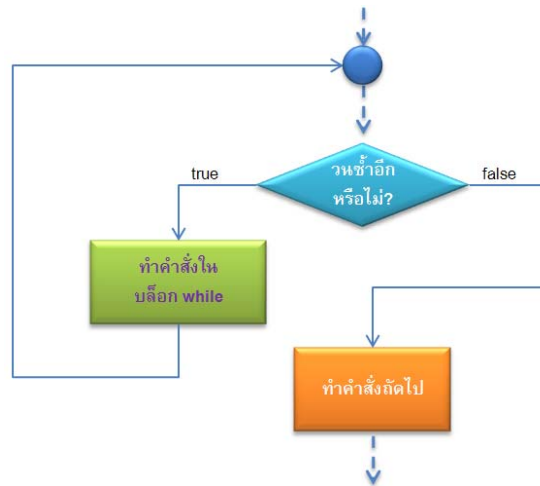
ผลลัพธ์



รูปที่ 4-7 : ผลลัพธ์ การใช้คำสั่ง for โดยเพิ่มครั้งละ 2

การวนซ้ำด้วยการใช้ while

การวนลูปด้วยการใช้ while เป็นคำสั่งที่มีการตรวจสอบเงื่อนไขก่อน แล้วจึงจะทำงาน



รูปที่ 4-8 : ผังการไหลคำสั่ง while

ในการใช้งาน while นั้นมีรูปแบบดังนี้

```

while (เงื่อนไข)
{
    < ขุดคำสั่ง >
}
  
```

ตัวอย่างที่ 4-7 การใช้ while

```

<?php
$i = 1;
while ($i <= 10) {
    echo $i++;
}
?>
  
```

การวนซ้ำด้วยการใช้ do-while

ปกติการใช้ while นั้นเราจะตรวจสอบเงื่อนไขก่อนการวนซ้ำในแต่ละครั้ง แต่ถ้าเราต้องการให้การตรวจสอบเงื่อนไขไปอยู่ด้านหลัง (คือต้องวนซ้ำอย่างน้อย 1 รอบก่อน) เราจะใช้ do มาช่วย โดยจะนำเงื่อนไขการตรวจสอบวนซ้ำ while ไปไว้ด้านหลัง ดังรูปแบบต่อไปนี้ ยังมีรูปแบบการวนซ้ำในลักษณะวนซ้ำด้วยจำนวนรอบที่ไม่แน่นอนแล้ว อีกหลายรูปแบบ ได้แก่

```
do
{
    < ทำงานตามคำสั่ง >
}
while (เงื่อนไข) ;
```

ตัวอย่างที่ 4-8 การใช้ do while

```
<?php
$i = 0;
do {
    echo $i;
} while ($i > 0);
?>
```

การประยุกต์ใช้งาน

การทำงานของโปรแกรม PHP จะใช้งานร่วมกับโปรแกรม HTML โดยการสร้างฟอร์มจาก HTML แล้วส่งให้ไฟล์ PHP ประมวลผล หรือทำการเก็บข้อมูลลงฐานข้อมูล ตรวจสอบความถูกต้อง ซึ่งโปรแกรม PHP มีคำสั่งเพื่อใช้ในการประมวลผลและตรวจสอบความถูกต้อง และสามารถ เก็บข้อมูลลงฐานข้อมูลได้

ตัวอย่างที่ 4-9 ทดสอบการส่งข้อมูล

```
1 <FORM METHOD=POST ACTION="testForm.php">
2 User Name ::
3 <INPUT TYPE="text" NAME="txtUser" SIZE="50" MAXLENGTH="10" ReadOnly value="Napatsarun"><BR>
4 Password :: <INPUT TYPE="password" NAME="txtPassword"><BR>
5 Address :: <TEXTAREA NAME="txtAddress" ROWS="3" COLS="50">กรุณกรอกที่อยู่</TEXTAREA><BR>
6 Country ::
7 <SELECT NAME="txtCountry">
8 <option value="USA">อเมริกา
9 <option value="THAI">ประเทศไทย
10 <option value="AUS">ออสเตรเลีย
11 <option value="KOREA">เกาหลี
12 </SELECT><BR>
13 เพศ ::
14 <INPUT TYPE="radio" NAME="txtGender" value="Male">ชาย
15 <INPUT TYPE="radio" NAME="txtGender" value="FeMale">หญิง
16 <BR>งานอดิเรก ::
17 <INPUT TYPE="checkbox" NAME="txtHobby[]" value="Movie">ดูหนัง
18 <INPUT TYPE="checkbox" NAME="txtHobby[]" value="Game">เล่นเกม
19 <BR>รูปภาพ :: <INPUT TYPE="file" Name="txtFile" size="30"><BR>
20 <INPUT TYPE="submit" VALUE="SEND">
21 <INPUT TYPE="reset" value="Cancel">
22 </FORM>
```

ผลลัพธ์

User Name ::

Password ::

Address ::

Country ::

เพศ :: ☒ ชาย ☐ หญิง

งานอดิเรก :: ☒ ดูหนัง ☐ เล่นเกมส์

รูปภาพ ::

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 แท็ก FORM เป็นแท็กกำหนดการเริ่มต้นของฟอร์ม โดยกำหนด วิธีการส่งข้อมูล เป็น POST และไฟล์ที่จะทำงานต่อไป คือไฟล์ testForm.php ไฟล์ใน Action ส่วนใหญ่จะเป็น PHP เพราะสามารถทำการคำนวณ งานต่าง ๆ ตรวจสอบเงื่อนไข ได้

บรรทัดที่ 2 เป็นการแสดงข้อความ User Name ::

บรรทัดที่ 3 เป็นการกำหนดให้เป็นการรับข้อมูลแบบ Text Box มีชื่อเป็น txtUser มีความยาวของ Text Box = 50 และกำหนดให้มีการตัดข้อมูลได้ ความยาว เท่ากับ 10 และสามารถอ่านได้อย่างเดียว โดยแสดงข้อความในช่อง เป็น Napatsarun

บรรทัดที่ 4 เป็นแสดงข้อความว่า Password :: และมีการรับข้อมูลแบบ Password มีลักษณะคล้าย Text Box แต่ การแสดงข้อความที่คีย์ จะแสดงเป็น ● แทนตัวอักษรที่คีย์

บรรทัดที่ 5 เป็นแสดงข้อความ Address :: และมีการรับข้อมูลแบบ Text Area คือเหมือนกับ Text Box แต่สามารถแสดงข้อความได้ หลายบรรทัด โดยกำหนดให้แสดง 3 บรรทัด และ 50 คอลัมน์

บรรทัดที่ 6 เป็นการแสดงข้อความ Country ::

บรรทัดที่ 7-12 เป็นคำสั่งสร้าง List Box โดยตั้งชื่อเป็น txtCountry และมีรายละเอียดใน List คือ อเมริกา ประเทศไทย ออสเตรเลีย และเกาหลี ตามลำดับ

บรรทัดที่ 13 เป็นการแสดงข้อความ เพศ

บรรทัดที่ 14-15 เป็นคำสั่ง เพื่อใช้สร้าง Radio Button คือ ปุ่มที่สามารถเลือกได้อย่างใดอย่างหนึ่ง เช่น เพศชาย หรือ เพศหญิง เลือกได้ เพศเดียว

บรรทัดที่ 16 เป็นการขึ้นบรรทัดใหม่ แล้วแสดงข้อความว่า งานอดิเรก

บรรทัดที่ 17-18 เป็นคำสั่ง เพื่อสร้าง Check Box โดยตั้งชื่อ txtHobby[] โดยตัวแปรนี้เก็บค่าเป็น Array เพราะ Check Box สามารถ เลือกได้หลายตัวเลือก

บรรทัดที่ 19 เป็นการขึ้นบรรทัดใหม่ แล้วแสดงข้อความว่า รูปภาพ และรับข้อมูลเป็นไฟล์

บรรทัดที่ 20 เป็นการสร้างปุ่ม Submit โดยแสดงข้อความบนปุ่มว่า SEND

บรรทัดที่ 21 เป็นการสร้างปุ่ม Reset โดยแสดงข้อความบนปุ่มว่า Cancel

บรรทัดที่ 22 เป็นการปิดฟอร์ม

ตัวอย่างที่ 4-10 ไฟล์ PHP ที่ทำงานต่อจากไฟล์ HTML

โปรแกรมนี้จะทำการแสดงข้อมูลที่ได้จากฟอร์ม HTML โดยมีการกำหนดเงื่อนไข คือ ถ้าเช็คเพศชายให้แสดงคำว่า MR และ เพศหญิงให้แสดง MISS ก่อนชื่อผู้ใช้ และข้อมูลประเทศให้แสดงเป็นภาษาไทย งานอดิเรก ให้แสดงทุกงานที่เลือก

```

1  <?
2  print "Hello ";
3  $txtUser=$_POST['txtUser'];
4  if($txtGender=="Male") print "MR."; else print "MISS ";
5  print $txtUser . "<br>";
6  print "Password :: " . $txtPassword . "<br>";
7  print "ที่อยู่ :: " . $txtAddress . "<br>";
8  print "ประเทศ :: ";
9  if($txtCountry=="USA")
10     print "อเมริกา ";
11  elseif ($txtCountry=="AUS")
12     print "ออสเตรเลีย ";
13  print "<br>งานอดิเรก มี " . Sizeof($txtHobby) . " อย่าง คือ ";
14  for ($i=0 ; $i < Sizeof($txtHobby) ; $i++)
15  {
16     # print $txtHobby[$i] . " ";
17     if ($txtHobby[$i]=="Movie")
18         print "ดูหนัง ";
19     elseif ($txtHobby[$i]=="Game")
20         print "เล่นเกม ";
21  }
22  ?>

```

ผลลัพธ์

Hello MR.Napatsarun
 Password :: 1234
 ที่อยู่ :: กรุงเทพฯ
 ประเทศ :: อเมริกา
 งานอดิเรก มี 2 อย่าง คือ ดูหนัง เล่นเกมส์

อธิบายรหัสคำสั่ง

- บรรทัดที่ 1 <? เป็นการเริ่มต้นเอกสาร PHP
- บรรทัดที่ 2 เป็นการแสดงข้อความ Hello
- บรรทัดที่ 3 เป็นการตั้งค่าตัวแปร โดยใช้ Method POST ส่งผ่านตัวแปร txtUser
- บรรทัดที่ 4 เป็นตรวจสอบเงื่อนไข ว่า txtGender เป็น Male ให้แสดงคำว่า MR ถ้าไม่ใช่ ให้แสดง MISS โดยคำสั่ง if สามารถเขียนได้ใน บรรทัดเดียวหาก การทำงานของคำสั่งไม่ซับซ้อน สามารถเขียนได้ด้วยคำสั่งเพียงคำสั่งเดียว
- บรรทัดที่ 5 เป็นการแสดงข้อความ ในตัวแปร ของ text Box ที่รับมาจาก User name ของฟอร์ม
- บรรทัดที่ 6 เป็นการแสดงข้อความ Password และแสดงรหัสลับที่กรอกเข้ามา
- บรรทัดที่ 7 เป็นการแสดงข้อความ ที่อยู่ และที่อยู่ที่ได้กรอกไว้ในฟอร์ม
- บรรทัดที่ 8 เป็นการแสดงข้อความ ประเทศ
- บรรทัดที่ 9-12 เป็นการตรวจสอบเงื่อนไข เพื่อแสดงประเทศ ที่เลือกจากฟอร์ม โดยตรวจสอบจากค่า Value ของ txtCountry
- บรรทัดที่ 13 เป็นการแสดงข้อความ งานอดิเรก และ จำนวนของข้อมูล ในตัวแปร txtHobby ซึ่งเป็นตัวแปร เกิดจากการ เช็คงานอดิเรกที่ฟอร์ม ก่อนหน้า ถ้าเลือก 2 งานจะมีค่าเท่ากับ 2
- บรรทัดที่ 14 เป็นคำสั่ง for loop เพื่อใช้ในการควบคุมการพิมพ์ข้อมูลของ ตัวแปร txtHobby
- บรรทัดที่ 15 , 21 เป็นการกำหนดของขอบเขตการทำงานของคำสั่ง for
- บรรทัดที่ 16 เป็นการค่าในตัวแปร txtHobby ออกมา แต่ Comment ไว้ จึงไม่มีผล
- บรรทัดที่ 17-20 เป็นการกรณีเหมือนกับการแสดงข้อมูลประเทศ เพราะการเก็บค่าตัวแปร กับการแสดงผล ไม่ตรงกัน จึงต้องตรวจสอบเพื่อแสดงผล

ตัวอย่างที่ 3.10 การแสดงวัน เดือน ปี โดยใช้ List Box

```

1 BirthDate ::
2 <SELECT NAME="txtDay">
3   <? for ($i=1 ; $i <=31 ; $i++) { ?>
4     <OPTION VALUE=<?=$i?>> <?=$i ?>
5     <? } ?>
6 </SELECT>
7 <? $mon=Array("Jan","Feb","Mar","Apr","May","Jun","Jul","Aug","Sep","Oct","Nov","Dec") ?>
8 <SELECT NAME="txtMonth">
9   <? for ($i=0 ; $i <Sizeof($mon); $i++) { ?>
10    <OPTION VALUE=<?=$mon[$i]?>> <?=$mon[$i] ?>
11    <? } ?>
12 </SELECT>
13 <SELECT NAME="txtYear">
14   <? for ($i=2450 ; $i <= 2550 ; $i++) { ?>
15     <OPTION VALUE=<?=$i?>> <?=$i ?>
16     <? } ?>
17 </SELECT>

```

ผลลัพธ์

BirthDate ::

1	Jan	2450
	Jan	
	Feb	
	Mar	
	Apr	
	May	
	Jun	
	Jul	
	Aug	
	Sep	
	Oct	
	Nov	
	Dec	

อธิบายรหัสคำสั่ง

บรรทัดที่ 3 เป็นการสร้างคำสั่ง for เพื่อวนลูป ตั้งแต่ 1 -31

บรรทัดที่ 4 เป็นการกำหนดค่าให้กับ ListBox โดยตัวแปร \$i ตัวแรก เป็นการกำหนดค่าให้กับ Value และ ตัวแปร \$i ตัวที่สอง เป็นการแสดงค่าวันที่ ใน List สำหรับ การใช้คำสั่ง <?=\$i?> คือ การแสดงค่าตัวแปร \$i ออกมา เนื่องด้วยมีเพียงคำสั่งเดียว ในเครื่องหมาย <? ?> จึงใช้ เครื่องหมาย = แทนคำสั่ง print ได้

บรรทัดที่ 7 เป็นการสร้างตัวแปร \$mon ให้เป็นตัวแปร Array ที่เก็บค่า เดือนไว้

บรรทัดที่ 9 เป็นการสร้างคำสั่ง for เพื่อวนลูป ตั้งแต่ 0 จนถึง จำนวนของเดือน -1 ก็คือ 11

บรรทัดที่ 10 เป็นการแสดงข้อมูลใน List เหมือนกับวันที่ แต่ค่าเดือนอยู่ในตัวแปร Array จึงต้องกำกับด้วย Index ว่าต้องการแสดงตัวที่เท่าไร

บรรทัดที่ 14 เป็นการสร้างคำสั่ง for เพื่อวนลูป ตั้งแต่ 2450-2550

ตัวอย่างที่ 4-11 ไฟล์ Login.html เพื่อเข้าระบบการสร้างข้อสอบและคำนวณคะแนน

```
1 <FORM METHOD=POST ACTION="quiz.php">
2 User ::<INPUT TYPE="text" NAME="txtUser"><BR>
3 Password ::<INPUT TYPE="Password" NAME="txtPassword"><BR>
4 <INPUT TYPE="submit" value="Log on">
5 </FORM>
```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นการสร้างฟอร์ม เพื่อกำหนดให้ไปทำงานต่อที่ไฟล์ quiz.php เมื่อกดปุ่ม submit

บรรทัดที่ 2 เป็นการแสดงข้อความ User และรองรับข้อความ User

บรรทัดที่ 3 เป็นการแสดงข้อความ Password และรองรับข้อความ เป็นแบบ Password

บรรทัดที่ 4 เป็นการสร้างปุ่ม Submit โดยมีข้อความว่า Log on

ผลลัพธ์

User ::

Password ::

ตัวอย่างที่ 3.12 ไฟล์ Quiz.php

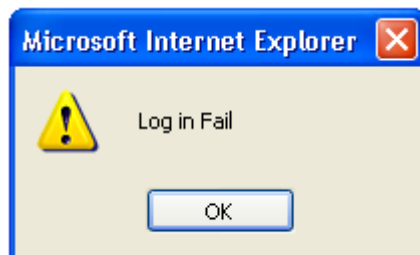
```

1 <!-- quiz.php -->
2 <? if ( ($txtUser!="Admin") || ($txtPassword!="123"))
3 print "<SCRIPT> alert('Log in Fail'); window.location='login.html';</SCRIPT>";
4 else ?>
5 <FORM METHOD=POST ACTION="calScore.php">
6 ชื่อ - นามสกุล <INPUT TYPE="text" NAME="txtName">
7 <BR>วันที่ทำข้อสอบ
8 <? $yearS = date("Y")+543; $monthS = date("M"); $dayS = date("d"); ?>
9 <SELECT NAME="txtDay">
10 <? for ($i=1; $i<=31; $i++) { ?>
11 <option value="<?=$i?>" <?if ($dayS==$i) echo "selected"; ?> ><?=$i?>
12 <? } ?> </SELECT>
13 <? $mon = Array("Jan","Feb","Mar","Apr","May","Aug","Oct","Nov","Dec") ?>
14 <SELECT NAME="txtMonth">
15 <? for ($i=0; $i<Sizeof($mon); $i++) { ?>
16 <option value="<?=$mon[$i]?>" <?if ($monthS==$mon[$i]) echo "selected"; ?> ><?=$mon[$i]?>
17 <? } ?> </SELECT>

18 <SELECT NAME="txtYear">
19 <? for ($i=2550; $i<=2560; $i++) { ?>
20 <option value="<?=$i?>" <?if ($yearS==$i) print "selected"; ?> ><?=$i?>
21 <? } ?> </SELECT><HR>
22 <B>ข้อ 1. HTML ย่อมาจากคำว่าอะไร <BR></B>
23 <INPUT TYPE="radio" NAME="ex1" value="1">
24 Hypertext Markup Language <BR>
25 <INPUT TYPE="radio" NAME="ex1" value="0">
26 High Text Markup Language <BR>
27 <INPUT TYPE="radio" NAME="ex1" value="0">
28 High Tech Make Language <BR>
29 <INPUT TYPE="radio" NAME="ex1" value="0">
30 Hyper Tech Markup Language <BR>
31 <HR>
32 <B>ข้อ 2. ไฟล์ PHP ต้อง Save ไว้ที่ใด <BR></B>
33 <INPUT TYPE="radio" NAME="ex2" value="0">
34 C:\inetpub\wwwroot <BR>
35 <INPUT TYPE="radio" NAME="ex2" value="0">
36 ที่ไหนก็ได้ <BR>
37 <INPUT TYPE="radio" NAME="ex2" value="1">
38 C:\AppServ\www <BR>
39 <INPUT TYPE="radio" NAME="ex2" value="0">
40 My Document <BR>
41 <HR>
42 <INPUT TYPE="submit" value="ส่งข้อสอบ">
43 </FORM>

```

ผลลัพธ์



รูปที่ 4-9 : แสดงผลลัพธ์การรันโปรแกรม QUIZ

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นการสร้างหมายเหตุในภาษา HTML

บรรทัดที่ 2-3 เป็นการตรวจสอบ user และ Password ว่าตรงกับที่กำหนดหรือไม่ ถ้าไม่ตรงให้แสดง แสดงข้อความ Login Fail แล้วไปทำงานที่ไฟล์ Login.html คือรับ User คือให้กลับไป Login ใหม่

บรรทัดที่ 8 เป็นคำสั่งเพื่อเก็บวันที่ ปัจจุบัน

บรรทัดที่ 9-20 เป็นการสร้างList Box วัน เดือน ปี โดยแสดง ค่าไว้ ณ วันที่ปัจจุบัน

บรรทัดที่ 22-30 เป็นการสร้างข้อสอบข้อที่ 1 กำหนดให้ ตัวเลือกที่ 1 ถูก มีค่าเป็น 1

บรรทัดที่ 32-40 เป็นการสร้างข้อสอบข้อที่ 2 กำหนดให้ ตัวเลือกที่ 3 ถูก มีค่าเป็น 1

ผลลัพธ์

ชื่อ - นามสกุล
วันที่ทำข้อสอบ

ข้อ 1. HTML ย่อมาจากคำว่าอะไร

- ☐ Hypertext Markup Language
- ☐ High Text Markup Language
- ☐ High Tech Make Language
- ☐ Hyper Tech Markup Language

ข้อ 2. ไฟล์ PHP ต้อง Save ไว้ที่ใด

- ☐ C:\Inetpub\wwwroot
- ☐ ที่ไหนก็ได้
- ☐ C:\AppServ\www
- ☐ My Document

ตัวอย่างที่ 4-12 ๗ไฟล์ CalScore.php

```
1 <?
2 print "คุณ ".$txtName."<br>";
3 $score = $ex1 + $ex2;
4 print "คุณได้คะแนน ".$score." คะแนน";
5 ?>
```

ผลลัพธ์

คุณ นภัสสรณ์ ขั้วवालานนท์
คุณได้คะแนน 2 คะแนน

อธิบายรหัสคำสั่ง

บรรทัดที่ 2 เป็นคำสั่งให้แสดงข้อความ ชื่อ
บรรทัดที่ 3 เป็นคำสั่งให้คำนวณคะแนน
บรรทัดที่ 4 เป็นคำสั่งให้แสดงคะแนน

อดีต....ไม่ขยัน ปัจจุบัน....ไม่ชวนขาย ไม่ต้องทนาย.....อนาคต
ทำวันนี้ให้ดีที่สุด

บทที่ 5 ฐานข้อมูล

ประมาณทศวรรษที่ 1970 องค์กรส่วนใหญ่มีการเก็บข้อมูลด้วยคอมพิวเตอร์ในลักษณะแฟ้มข้อมูล (File System) โดยได้ทำการเก็บข้อมูลที่ระบบหรือหน่วยงาน ดังนั้น ระบบข้อมูลขององค์กร ในแต่ละ ระบบจึงเป็นอิสระต่อกัน ทำให้แต่ละระบบหรือหน่วยงานมีข้อมูลของตนเองโดยไม่เชื่อมโยงกัน ซึ่งสถาน การณ์นี้เรียกว่า การจัดเก็บแฟ้มแบบดั้งเดิม (Traditional File Environment) หรือเรียกระบบนี้ว่า ระบบ แฟ้มข้อมูล (File System)

ปัญหาแฟ้มข้อมูล

- ความซ้ำซ้อนของข้อมูล (Data Redundancy)
- ความไม่สอดคล้องกันของข้อมูล (Data Inconsistency)
- ขาดความยืดหยุ่น (Lack of Flexibility)
- ความไม่ปลอดภัยของข้อมูล (Poor Security)
- ขาดความเป็นอันหนึ่งอันเดียวกันของข้อมูล (Lack of Data Integrity)

ปัญหาแฟ้มข้อมูล

- ข้อมูลมีความสัมพันธ์ลักษณะขึ้นต่อกันกับโปรแกรม (Application/Data Dependencies)
- ข้อมูลแยกอิสระต่อกัน (Data Isolation)
- ขาดการใช้ข้อมูลร่วมกัน (Lack of Data Sharing)

ฐานข้อมูลและการจัดการฐานข้อมูล (Databases and Databank Management)

ความหมายของฐานข้อมูล

ฐานข้อมูล (Database หรือ Databank) คือการจัดกลุ่มของแฟ้มข้อมูล ที่มีความสัมพันธ์กันเพื่อนำไปใช้ในการทำงาน โดยใช้ซอฟต์แวร์ชุดหนึ่ง (DBMS) ซึ่งสามารถเข้าถึงข้อมูลในฐานข้อมูลได้ ซึ่งจะช่วยบรรเทาปัญหาต่าง ๆ ที่เกิดจากแฟ้มข้อมูลได้

ฐานข้อมูลและการจัดการฐานข้อมูล

ฐานข้อมูลมีหลายประเภท เช่น ฐานข้อมูลทางภูมิศาสตร์ (Geographical information database), ฐานข้อมูลความรู้ (Knowledge database), ฐานข้อมูลมัลติมีเดีย (Multimedia database) เป็นต้น

ระบบการจัดการฐานข้อมูล (Database Management Systems – DBMS) คือ ซอฟต์แวร์หรือกลุ่มของโปรแกรมที่ช่วยในการวางแผน รวบรวมข้อมูล จัดการและเข้าถึงข้อมูล ได้อย่างมีประสิทธิภาพ ผู้ใช้สามารถใช้ DBMS ในการเพิ่มเติมข้อมูล ลบข้อมูล แสดงผล พิมพ์ ค้นหา เลือก จัดเรียง หรือยกระดับของข้อมูลได้

ประเภทของ DBMS

DBMS มีหลายประเภท ตั้งแต่เป็นโปรแกรมที่ใช้กับไมโครคอมพิวเตอร์ จนถึงโปรแกรมที่ใช้กับเมนเฟรม นอกจากนี้ข้อมูลที่จัดการโดย DBMS ยังสามารถเก็บข้อมูลที่เป็นรูปภาพ ฟิคเสียง และรูปภาพได้ด้วย

ส่วนประกอบของ DBMS มี 4 ส่วนหลัก ๆ คือ

1. **โมเดลของข้อมูล (Data Model)** ทำหน้าที่กำหนดรูปแบบของโครงสร้างของข้อมูล เช่น จะเป็นลำดับชั้น (hierarchy) หรือ แบบเครือข่าย (network) หรือ แบบความสัมพันธ์ (relational)
2. **ภาษาคำจำกัดความของข้อมูล (Data Definition Language – DDL)** เป็นการกำหนดลักษณะของข้อมูลในแต่ละเรคคอร์ดหรือฟิลด์ที่ปรากฏในฐานข้อมูล เช่น จะตั้งชื่อว่าอย่างไร เป็นข้อมูลชนิดไหน ความยาวเรคคอร์ดเท่าใด รวมทั้งความสัมพันธ์ระหว่างเรคคอร์ดต่าง ๆ และลักษณะของคีย์
3. **ภาษาในการจัดการข้อมูล (Data Manipulation Language – DML)** เป็นภาษาที่ใช้ในการถามเกี่ยวกับเนื้อหาในฐานข้อมูล หรือใช้เป็นการเก็บหรือปรับปรุงข้อมูลให้ทันสมัย และการพัฒนา application นอกจากนี้ยังช่วยให้ผู้ใช้สามารถดึง จัดเรียง แสดงผล ลบข้อมูล ในฐานข้อมูลได้ด้วย ภาษาที่ใช้ในการจัดการข้อมูล อาจจะใช้ภาษา ยุคที่ 3 ยุคที่ 4 หรือ Object-Oriented ก็ได้ ตัวอย่าง คือ Structured Query Language (SQL)
4. **พจนานุกรมข้อมูล (Data Dictionary)** เป็นการเก็บรวบรวมคำจำกัดความของข้อมูล และลักษณะข้อมูลต่าง ๆ ที่อยู่ในฐานข้อมูล อันจะทำให้เกิดมาตรฐานความสอดคล้องของข้อมูลในแฟ้มต่าง ๆ และยังทำให้การพัฒนาโปรแกรมทำได้รวดเร็ว เพราะโปรแกรมเมอร์ สามารถดูข้อมูลจากพจนานุกรมข้อมูลได้

องค์ประกอบของฐานข้อมูล มีดังต่อไปนี้

- 1) **เอนติตี้ (Entity)** เป็นเรื่องเกี่ยวกับบุคคล สถานที่ สิ่งของ หรือเหตุการณ์ที่ต้องการเก็บข้อมูล
- 2) **ลักษณะเฉพาะของเอนติตี้ (Data items หรือ Attribute)** คือลักษณะของเอนติตี้ที่ต้องการเก็บข้อมูล เช่น เอนติตี้ของนักศึกษาประกอบด้วย attribute คือ รหัสนักศึกษา, ชื่อ, สกุล, คณะ, กลุ่ม ฯลฯ

3) **ระเบียบหรือเรคคอร์ด (Records)** คือ ชุดของลักษณะเฉพาะที่เกี่ยวข้องกับเอนทิตีหนึ่ง ๆ ซึ่งจะใช้ในการประมวลผลด้วยกัน

4) **แฟ้มข้อมูล (File)** ประกอบด้วยเรคคอร์ดที่สัมพันธ์กันหลาย ๆ อันมารวมกัน เช่น แฟ้มข้อมูลพนักงานจะประกอบด้วยเรคคอร์ดของพนักงานแต่ละคน

5) **ฐานข้อมูล (Database)** ประกอบด้วยแฟ้มข้อมูลหลาย ๆ แฟ้มที่มีความสัมพันธ์กันมารวมกัน

ความสัมพันธ์ของข้อมูลแบ่งเป็น 3 ประเภทคือ

1) **ความสัมพันธ์แบบ One to One** คือความสัมพันธ์ของข้อมูล 2 ตัว ที่มีลักษณะ 1 ต่อ 1 หรือข้อมูลตัวหนึ่ง จะมีความสัมพันธ์กับข้อมูลอีกตัวหนึ่งได้เพียงค่าเดียวเท่านั้น

2) **ความสัมพันธ์แบบ One to Many** คือ ความสัมพันธ์ซึ่งข้อมูลตัวหนึ่งมีความสัมพันธ์กับข้อมูลตัวอื่นได้หลายอย่าง

3) **ความสัมพันธ์แบบ Many to Many** คือ ความสัมพันธ์ซึ่งข้อมูลตัวหนึ่งมีหลายค่า และมีความสัมพันธ์กับข้อมูลตัวอื่นได้หลายอย่าง เช่น มีวิชาที่เปิดสอนหลายวิชา แต่ละวิชามีนักศึกษาหลายคน

ประเภทของการออกแบบฐานข้อมูล

1. ฐานข้อมูลแบบลำดับชั้น(Hierarchical Database Model)

ลักษณะเป็นลำดับชั้นคล้ายคลึงกับความสัมพันธ์ของโครงสร้างสายการบังคับบัญชาที่พบในองค์กรทั่วไป โดยมีลำดับชั้นลดหลั่นกันลงมาเป็นชั้นๆ ข้อมูล ที่ใช้ฐานข้อมูลลำดับชั้นจะต้องเป็นข้อมูลที่มีความสัมพันธ์แบบ one to one หรือ one to many เท่านั้น

ข้อดี

มีประสิทธิภาพในการค้นหา สามารถ ทำได้รวดเร็วเนื่องจากจัดลำดับชั้นและความสัมพันธ์เอาไว้แล้วทำให้ไม่เสียเวลา ในการค้นหาข้อมูลที่ไม่จำเป็น

ข้อจำกัด

ต้องมีการกำหนดลักษณะความสัมพันธ์ของข้อมูลทุกตัวก่อนเพื่อนำมาจัดเป็นโครงสร้างได้ และการค้นหาข้อมูลมีจำกัด นอกจากนี้โครงสร้างแบบนี้ไม่ได้เหมาะกับงานทุกประเภท

2. ฐานข้อมูลแบบเครือข่าย(Network Database Model)

คล้ายกับฐานข้อมูลแบบลำดับชั้น แต่ข้อมูลมีความสัมพันธ์กันในลักษณะ many to many เท่านั้น

ข้อดี

ข้อมูลที่อยู่ในเครือข่ายไม่มีข้อจำกัดเรื่องความสัมพันธ์ มีความสอดคล้องในโลกของความเป็นจริงมากกว่า

ข้อจำกัด

มีความซับซ้อนมาก โดยเฉพาะถ้ามีการเพิ่มลบข้อมูลและความสัมพันธ์ การทำงานจะซับซ้อนมากและการบำรุงรักษาจะยากขึ้นตามไปด้วย

3. ฐานข้อมูลแบบสัมพันธ์(Relational Database Model)

ใช้ตารางเพื่อเก็บข้อมูลให้เป็นระเบียบ ตารางเหล่านี้เรียกว่า ตารางความสัมพันธ์ (relations) การออกแบบฐานข้อมูลแบบความสัมพันธ์จะประกอบด้วยตารางที่มีขนาดไม่ใหญ่มากนักหลายๆ ตาราง แต่ละแถวแนวนอนเรียกว่า row ในตารางจะบรรจุข้อมูลเป็นชุดๆ เรียกว่า record ข้อมูล 1 ชุดในแต่ละชุดเรียกว่า field

หลักการพื้นฐานในการสร้าง RDB

- การเรียงลำดับก่อนหลังของเซลล์ในแนวตั้ง column หรือแนวนอน row ไม่มีความสัมพันธ์ต่อการค้นหาข้อมูล
- ในแต่ละตารางต้องมีตัวชี้วัดความสัมพันธ์ key ที่เชื่อมโยงกับตารางอื่นๆ
- ต้องไม่มีความซ้ำซ้อนในแต่ละตาราง
- ค่าของข้อมูลในแต่ละเซลล์มีได้ค่าเดียวเท่านั้น

ฐานข้อมูลแบบความสัมพันธ์

ข้อดี

- ✓ เป็นแนวคิดที่ง่าย และสามารถเชื่อมโยงข้อมูลได้โดยไม่จำเป็นต้องทราบความสัมพันธ์ของข้อมูลล่วงหน้า
- ✓ มีความยืดหยุ่นต่อผู้ใช้สูง และสามารถนำไปใช้กับแอปพลิเคชันได้หลายอย่าง
- ✓ มองเห็นโครงสร้างข้อมูลได้ง่าย โดยพิจารณาจากตาราง

ข้อจำกัด

- ถ้าฐานข้อมูลมีขนาดใหญ่มากจะประกอบด้วยตารางความสัมพันธ์จำนวนมาก ทำให้การออกแบบมีความซับซ้อน และทำให้การค้นหาข้อมูลหรือการดึงข้อมูลมีความล่าช้า
- ผลจากการค้นหาและการดึงข้อมูลที่ซ้ำทำให้การประมวลผลไม่มีประสิทธิภาพทำให้เกิดการไม่ยอมรับระบบฐานข้อมูลแบบนี้
- มีความเป็นอันหนึ่งอันเดียวกันของข้อมูลน้อย เมื่อเปรียบเทียบกับแบบลำดับชั้นหรือแบบเครือข่าย

แนวโน้มของฐานข้อมูล

- Object-Oriented Database Model
- Hypermedia
- Data Warehouse
- Data Mining

Object-Oriented Database Model

OODB หรือ O-O Database Model เป็นการจัดการข้อมูลโดยการเก็บทั้งข้อมูลและวิธีการจัดการข้อมูลไว้ในอ็อบเจกต์(Object) ซึ่งสามารถดึงและใช้งานร่วมกันได้โดยอัตโนมัติ

องค์ประกอบที่สำคัญของ OODB

- **อ็อบเจกต์(Object):** เป็นข้อมูลจำนวนมากนักที่นำมารวมกันมีความหมายเหมือนกับแอนติตี้ ซึ่งเป็นตัวแทนของคน สถานที่ สิ่งของ แต่อ็อบเจกต์จะรวมถึงกระบวนการหรือวิธีการที่เกี่ยวข้องกับการประมวลผลข้อมูลด้วย
- **แอตทริบิวต์(Attribute):** เป็นลักษณะของ อ็อบเจกต์ ในช่วงเวลาหนึ่งๆ เช่น อายุของพนักงาน
- **วิธีการ(Method):** หรือพฤติกรรมของอ็อบเจกต์ เมื่อเกิดการปฏิบัติการขึ้นจะมีการส่งข้อมูลไปยังอ็อบเจกต์ที่ส่งมา เพื่อจะกระตุ้นให้เกิดปฏิบัติการอื่นที่ต่อเนื่องกัน

Hypermedia database

Hypermedia database เป็นการจัดการข้อมูลในลักษณะเหมือนกับเครือข่ายของโหนด แต่ละโหนดจะประกอบด้วยข้อมูลซึ่งจะเป็นข้อความ รูปภาพ เสียง ภาพเคลื่อนไหว หรือโปรแกรมการทำงานอื่นๆ ฐานข้อมูลแบบ OODB และ Hypermedia จะสามารถเก็บข้อมูลที่มีลักษณะซับซ้อนมากกว่าฐานข้อมูลแบบตาราง ประสิทธิภาพจะดีกว่าแบบความสัมพันธ์หากมีข้อมูลจำนวนมาก

ดาต้าแวร์เฮาส์(Data Warehouse)

ดาต้าแวร์เฮาส์(Data Warehouse) เป็นฐานข้อมูลที่เก็บข้อมูลทั้งในปัจจุบันและในอดีต ซึ่งดึงมาจากระบบปฏิบัติการหลายระบบ และนำมารวมกันเพื่อประโยชน์ในการจัดทำรายงานหรือวิเคราะห์ข้อมูล ดาต้าแวร์เฮาส์ประกอบด้วยเครื่องมือในการถามที่เป็นมาตรฐาน (standardized query tool) เครื่องมือในการวิเคราะห์ และเครื่องมืออำนวยความสะดวกในการทำงานในลักษณะกราฟิก ดาต้าแวร์เฮาส์สามารถทำการวิเคราะห์ข้อมูลในเชิงวิเคราะห์แนวโน้มหรือเจาะหาข้อมูล (drill) ในรายละเอียดเมื่อต้องการได้

Data mart หมายถึงดาต้าแวร์เฮาส์ขนาดเล็ก ซึ่งประกอบด้วยข้อมูลขององค์กรบางส่วน สำหรับผู้ใช้กลุ่มใดกลุ่มหนึ่งโดยเฉพาะ

ลักษณะที่สำคัญของดาต้าแวร์เฮาส์

- ข้อมูลมาจากฐานข้อมูลหลายแห่ง
- ดาต้าแวร์เฮาส์มีหลายมิติ
- ดาต้าแวร์เฮาส์ใช้สนับสนุนการตัดสินใจ ไม่ใช่ประมวลผลรายการ

ดาต้าไมนนิ่ง(Data Mining)

ดาต้าไมนนิ่งเป็นเครื่องมือของซอฟต์แวร์ที่ใช้ในการวิเคราะห์สารสนเทศโดยอัตโนมัติ เพื่อค้นหารูปแบบและความสัมพันธ์ของข้อมูลในดาต้าแวร์เฮาส์รวมทั้งพยากรณ์แนวโน้มและพฤติกรรมในอนาคต (Stair&Reynolds,1999:Turban, et al.,(2001))

จุดมุ่งหมายของดาต้าไมนนิ่ง

- การดึงรูปแบบ แนวโน้มและกฎเกณฑ์จากข้อมูลในดาต้าแวร์เฮาส์เพื่อที่จะประเมินกลยุทธ์ของหน่วยงาน
- ปรับปรุงความได้เปรียบในการแข่งขัน
- เป็นวิธีการที่นำมาใช้ในด้านการตลาด เช่น การรักษาลูกค้า การจัดโฆษณา ช่องทางการตลาด การวิเคราะห์ราคา

ส่วนประกอบของดาต้าไมนนิ่ง

- เครื่องมือในการถามและจัดทำรายงาน(Query-and-reporting-tools)
- อุปกรณ์ด้านปัญญาประดิษฐ์(Intelligent Agents)
- เครื่องมือในการวิเคราะห์ข้อมูลหลายมิติ(Multidimensional analysis tools-MDA)

ประเด็นการบริหารเกี่ยวกับดาต้าแวร์เฮาส์

ทุกคนหรือทุกองค์กรต้องการดาต้าแวร์เฮาส์หรือไม่ เนื่องจากมีเหตุผลด้านดาต้าแวร์เฮาส์ดังนี้

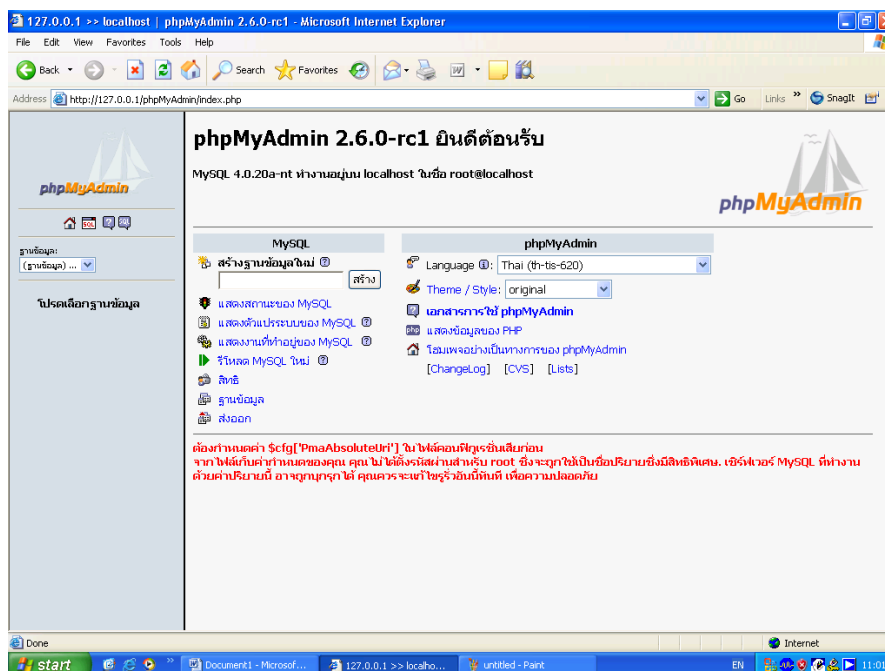
- ค่าใช้จ่ายค่อนข้างสูง
- บางหน่วยงานไม่จำเป็นต้องมีดาต้าแวร์เฮาส์
- หน่วยงานที่ดูแลด้านคอมพิวเตอร์มากพอในการพัฒนาหรือไม่
- หลายองค์กรมีดาต้าแวร์เฮาส์อยู่แล้ว
- ผู้ใช้เป็นใคร?
- สารสนเทศจะต้องได้รับการปรับปรุงให้ทันสมัยมากน้อยเพียงไร

ประเด็นการบริหารอื่น ๆ ของฐานข้อมูล

- ฐานข้อมูลแบบใดมีความเหมาะสมที่สุด
- ใครควรเป็นผู้ดูแลสารสนเทศขององค์กร
- ฐานข้อมูลและแอปพลิเคชันของฐานข้อมูลควรได้รับการพัฒนาและบำรุงรักษาอย่างไร
- ใครเป็นเจ้าของสารสนเทศ
- จริยธรรมในการจัดการสารสนเทศคืออะไร

การสร้างฐานข้อมูล MySQL

ฐานข้อมูล เป็นส่วนหนึ่งของระบบสารสนเทศที่จำเป็นต้องเก็บข้อมูลและสารสนเทศต่าง ๆ เก็บไว้เพื่อนำมาประมวลผล หรือนำมาแสดง คำนวณงานต่าง ๆ ฐานข้อมูลที่ใช้กับโปรแกรมที่นิยมใช้มากที่สุด คือ ฐานข้อมูล MySQL ซึ่งการสร้างฐานข้อมูลสามารถ สร้างได้จากโปรแกรม Internet Explorer โดยพิมพ์ พิมพ์คำว่า `http://127.0.0.1/phpMyAdmin/index.php` ลงในช่อง Address จะได้ผลลัพธ์ดังต่อไปนี้



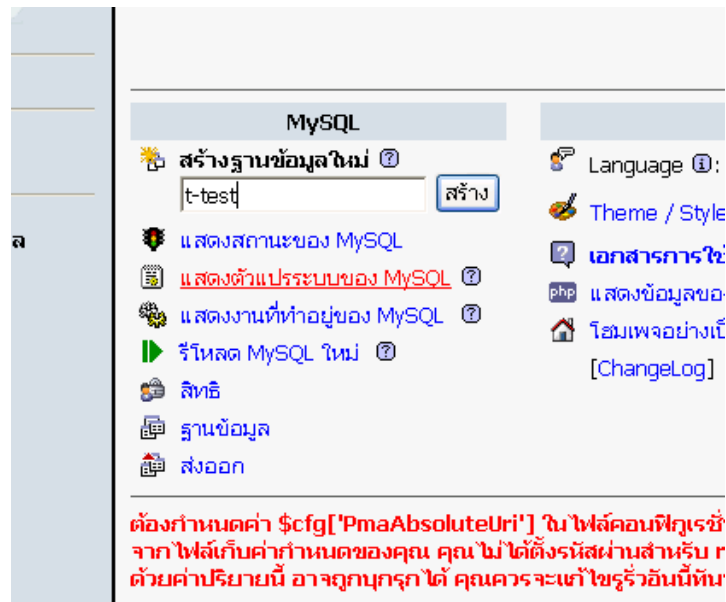
รูปที่ 5-1 : แสดงหน้าจอฐานข้อมูล MySQL

การสร้างตาราง

วิธีการสร้างตาราง

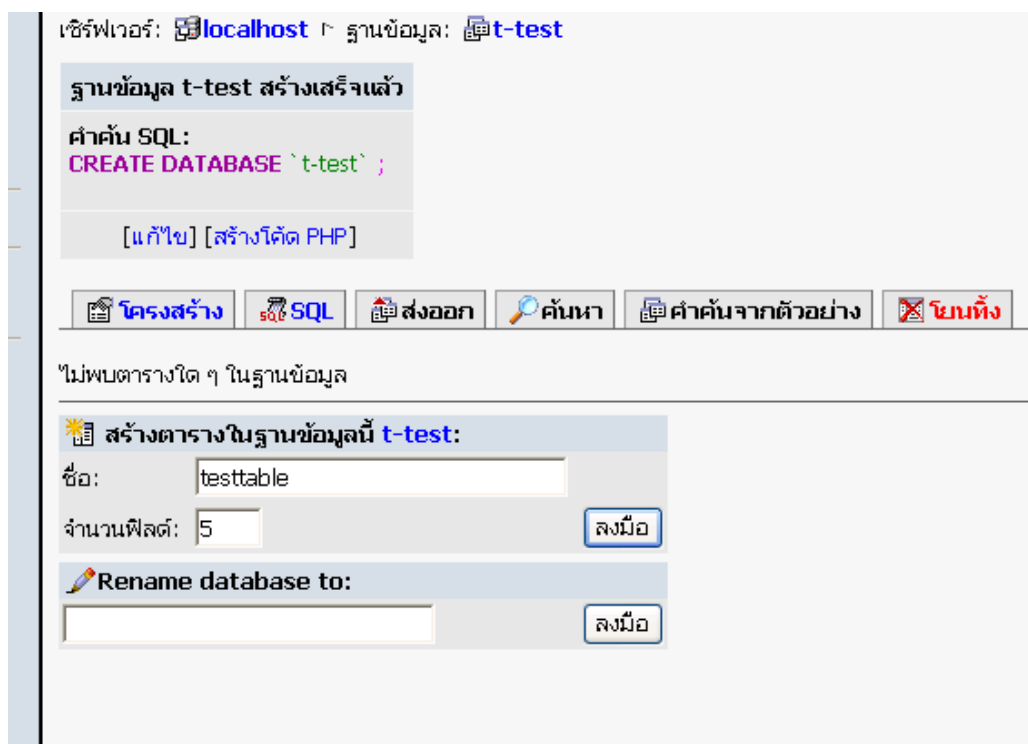
1. ใส่ชื่อฐานข้อมูลที่เราจะสร้างลงช่อง “สร้างฐานข้อมูลใหม่” แล้วคลิกปุ่ม “สร้าง”
2. เลือกฐานข้อมูลที่เราสร้างได้
3. ใส่ชื่อตารางที่ต้องการสร้าง และจำนวนฟิลด์ ที่ต้องการ
4. ใส่ชื่อฟิลด์ และรายละเอียดของฟิลด์

ขั้นตอนที่ 1 ใส่ชื่อฐานข้อมูลที่เราจะสร้างลงช่อง “สร้างฐานข้อมูลใหม่” แล้วคลิกปุ่ม “สร้าง”



รูปที่ 5-2: แสดงหน้าจอสร้างฐานข้อมูล

ขั้นตอนที่ 2 สร้างตาราง แล้วกำหนดฟิลด์ของตารางได้ดังรูป



รูปที่ 5-3 : แสดงหน้าจอสร้างชื่อตาราง

ขั้นตอนที่ 3 ใส่ชื่อฟิลด์ และกำหนดรายละเอียดในแต่ละฟิลด์

ฟิลด์	ชนิด	ความยาว/เขต*	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย**	เพิ่มเติม
	VARCHAR	50		not null		
	VARCHAR	50		not null		
	VARCHAR	50		not null		
	VARCHAR	50		not null		
	VARCHAR	50		not null		

หมายเหตุของตาราง: ชนิดตาราง:

รูปที่ 5-4 : แสดงหน้าจอสร้างฟิลด์

หมายเหตุ ไม่สามารถสร้างชื่อฐานข้อมูลซ้ำกันได้

ตัวอย่างที่ 5-1 สร้างฐานข้อมูล Employee

ขั้นตอนที่ 1 สร้างฐานข้อมูลชื่อ OA

รูปที่ 5-5 : แสดงหน้าจอสร้างฐานข้อมูล OA

ขั้นตอนที่ 2 สร้างตารางชื่อ employee มีจำนวนฟิลด์ 5 ฟิลด์

รูปที่ 5-6 : แสดงหน้าจอสร้างตารางพนักงาน

ขั้นตอนที่ 3 กำหนดรายละเอียดของฟิลด์ ที่ใช้เก็บข้อมูล กำหนดฟิลด์ ที่เป็น Primary Key แล้วกดปุ่ม บันทึก

Server: localhost Database: oa Table: employee

Field	Type	Length/Values	Attributes	Null	Default	Extra
emp_id	INT			not null		auto_increment
emp_name	VARCHAR	20		not null		
emp_surname	VARCHAR	20		not null		
emp_dept	VARCHAR	15		not null		
emp_salary	DOUBLE			not null		

รูปที่ 5-7 : แสดงหน้าจอการกำหนดรายละเอียดตารางพนักงาน

จะได้ ฐานข้อมูล เพื่อเก็บรายละเอียดของข้อมูล Employee ดังนี้

Server: localhost Database: oa Table: employee

Table employee has been created.

SQL-query:

```
CREATE TABLE `employee` (
  `emp_id` INT NOT NULL AUTO_INCREMENT ,
  `emp_name` VARCHAR(20) NOT NULL ,
  `emp_surname` VARCHAR(20) NOT NULL ,
  `emp_dept` VARCHAR(15) NOT NULL ,
  `emp_salary` DOUBLE NOT NULL ,
  PRIMARY KEY (`emp_id`)
);
```

[Edit] [Create PHP Code]

Structure Browse SQL Search Insert Export Operations Empty Drop

Field	Type	Attributes	Null	Default	Extra	Action
☐ emp_id	int(11)		No		auto_increment	
☐ emp_name	varchar(20)		No			
☐ emp_surname	varchar(20)		No			
☐ emp_dept	varchar(15)		No			
☐ emp_salary	double		No	0		

รูปที่ 5-8 : แสดงหน้าจอผลการสร้างตารางพนักงาน

ขั้นตอนที่ 4 กดปุ่ม แทรก เพื่อใส่ข้อมูล

โครงสร้าง เปิดดู SQL ค้นหา แทรก ส่งออก

แสดงระเบียนที่ 0 - 1 (2 ทั้งหมด, ค่าค้นหาใช้เวลา 0.0005 วินาที)

คำสั่ง SQL:

```
SELECT *
FROM `employee` LIMIT 0 , 30
```

[แก้ไข] [อธิบาย SQL] [สร้างโค้ด PHP] [Refresh]

แสดง : 30 แถว เริ่มจากแถวที่ 0

อยู่ใน และซ้ำหัวแถวทุกๆ เซลล์

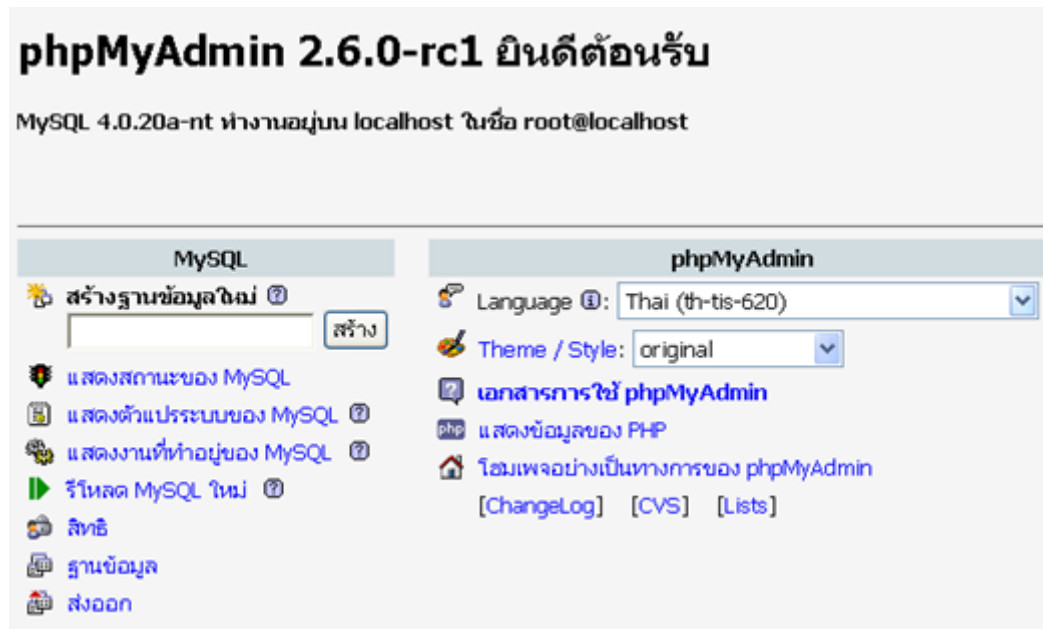
Sort by key:

	emp_id	emp_name	emp_surname	emp_dept	emp_salary
☐	5	สมชาย	แซ่เฮง	บัญชี	25000
☐	4	ดร.ณี	ชัชวาลานนท์	คอมพิวเตอร์	8500

รูปที่ 5-9 : แสดงหน้าจอผลการสร้างตารางพนักงาน

การตั้งค่า ภาษาไทย

หากข้อมูลเป็นภาษาไทย ต้องตั้งค่า ภาษาไทย ที่หน้าแรก แล้วเลือก Language เป็น Thai(th-tis-620) ดังภาพ



รูปที่ 5-10 : แสดงหน้าจอการตั้งค่ารหัสภาษาไทย

การสร้างแบบสอบถาม SQL Query

การสร้างแบบสอบถามจะสร้างที่ SQL และรูปแบบคำสั่ง SQL จะมาตรฐานของภาษา SQL คล้ายกับ โปรแกรมจัดการฐานข้อมูลอื่น ๆ

รูปแบบคำสั่ง แบบสอบถาม

```
SELECT ชื่อฟิลด์ 1 , ชื่อฟิลด์ 2
FROM ชื่อตาราง
WHERE เงื่อนไข
```

รูปแบบคำสั่ง การเพิ่มข้อมูล

```
INSERT INTO ชื่อตาราง VALUES( ค่าที่1, ค่าที่ 2, ค่าที่ 3, ...)
```

รูปแบบคำสั่ง การแก้ไขข้อมูล

```
UPDATE ชื่อตาราง SET ชื่อฟิลด์ 1= ค่าที่ต้องการแก้ไข , ชื่อฟิลด์ 2= ค่าที่ต้องการแก้ไข ,...
WHERE เงื่อนไข
```

รูปแบบคำสั่งการลบข้อมูล

```
DELETE FROM ชื่อตาราง
WHERE เงื่อนไข
```

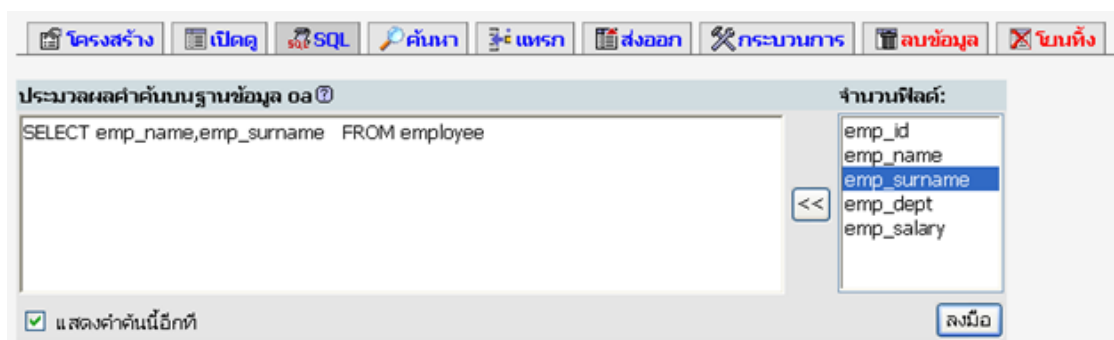
การสร้างแบบสอบถาม

SELECT ตามด้วยชื่อฟิลด์ที่ต้องการแสดง หากต้องการแสดง มากกว่า 1 ฟิลด์ ใช้เครื่องหมาย comma คั่นระหว่าง ชื่อฟิลด์ ถ้าต้องการแสดงทุกฟิลด์ ใช้ เครื่องหมาย * แทนแสดงทุกฟิลด์

FROM ตามด้วยชื่อ ตาราง

ตัวอย่างที่ 5-2 การแสดงข้อมูล ชื่อ และนามสกุล จาก ตาราง employee

```
SELECT emp_name , emp_surname
FROM employee
```



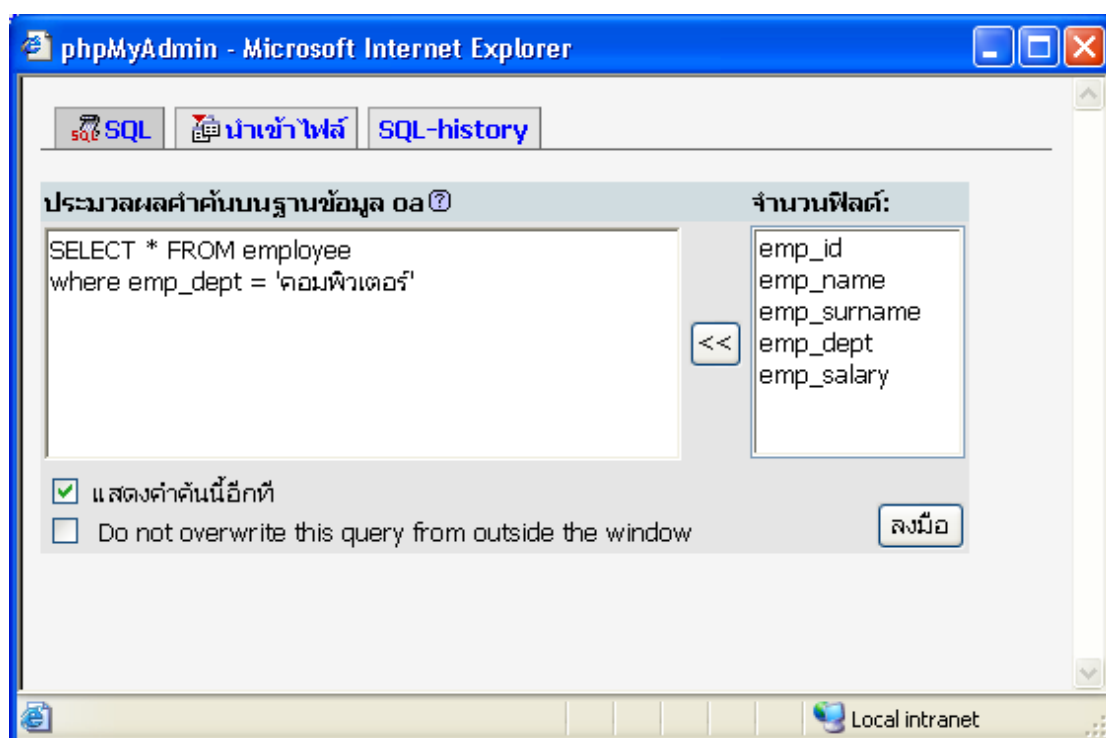
รูปที่ 5-11 : แสดงหน้าจอการสร้างแบบสอบถาม

อธิบายรหัสคำสั่ง

คำสั่ง SELECT จะตามด้วย ชื่อฟิลด์ ต้องการ ชื่อ และนามสกุล ใช้ emp_name , emp_surname

คำสั่ง FROM จะตามด้วย ชื่อตาราง คือ employee

ตัวอย่างที่ 5-3 แสดงข้อมูลพนักงาน แผนก คอมพิวเตอร์



รูปที่ 5-12 : แบบสอบถามแสดงพนักงานแผนกคอมพิวเตอร์

อธิบายรหัสคำสั่ง

คำสั่ง SELECT จะตามด้วย ชื่อฟิลด์ ต้องการทุกฟิลด์ ใช้ * ดอกจัน

คำสั่ง FROM จะตามด้วย ชื่อตาราง คือ employee

คำสั่ง WHERE จะตามด้วยเงื่อนไข โดยมี ชื่อฟิลด์ เปรียบเทียบ ค่า เช่น emp_dept = 'คอมพิวเตอร์' หมายความว่า ต้องการแสดงข้อมูลพนักงานแผนกคอมพิวเตอร์

ตัวอย่างที่ 5-4 แสดงข้อมูลพนักงาน ที่มีเงินเดือนมากกว่า 20,000

แสดงระเบียนที่ 0 - 0 (1 ทั้งหมด, ค่าค้นใช้เวลา 0.0005 วินาที)

คำสั่ง SQL:
SELECT *
FROM employee
WHERE emp_salary > 20000 **LIMIT** 0 , 30

[แก้ไข] [อธิบาย SQL] [สร้างโค้ด PHP] [Refresh]

แสดง : 30 แถว เริ่มจากแถวที่ 0

อยู่ใน **แนวนอน** และอ่านหัวแถวทุกๆ 100 เซลล์

	emp_id	emp_name	emp_surname	emp_dept	emp_salary
	5	สมชาย	แซ่สง	บัญชี	25000

รูปที่ 5-13 : แบบสอบถามแสดงพนักงานที่เงินเดือนมากกว่า 20000

อธิบายรหัสคำสั่ง

คำสั่ง **SELECT** จะตามด้วย ชื่อฟิลด์ ต้องการทุกฟิลด์ ใช้ * ดอกจัน

คำสั่ง **FROM** จะตามด้วย ชื่อตาราง คือ employee

คำสั่ง **WHERE** จะตามด้วยเงื่อนไข โดยมี ชื่อฟิลด์ เปรียบเทียบ ค่า เช่นกรณีนี้ คือ emp_salary > 20000 หมายความว่า ต้องการแสดงข้อมูลพนักงานที่มีเงินเดือนมากกว่า 20000

คำสั่ง **LIMIT** จะเริ่มต้นที่ record ที่เท่าไร แล้วนำมากี่ record กรณีนี้ Limit 0,30 คือ ต้องการตั้งแต่ record แรก แล้วนำมาแสดง 30 record ถ้ามี

ตัวอย่างที่ 5-5 แสดงข้อมูลพนักงานแผนกบัญชี ที่มีเงินเดือนมากกว่า 20,000

แสดงระเบียนที่ 0 - 0 (1 ทั้งหมด, ค่าค้นใช้เวลา 0.0006 วินาที)

คำสั่ง SQL:
SELECT *
FROM employee
WHERE emp_dept = 'บัญชี'
AND emp_salary > 20000 **LIMIT** 0 , 30

[แก้ไข] [อธิบาย SQL] [สร้างโค้ด PHP] [Refresh]

แสดง : 30 แถว เริ่มจากแถวที่ 0

อยู่ใน **แนวนอน** และอ่านหัวแถวทุกๆ 100 เซลล์

	emp_id	emp_name	emp_surname	emp_dept	emp_salary
	5	สมชาย	แซ่สง	บัญชี	25000

รูปที่ 5-14 : แบบสอบถามแสดงพนักงานแผนกบัญชีที่เงินเดือนมากกว่า 20000

อธิบายรหัสคำสั่ง

คำสั่ง **SELECT** จะตามด้วย ชื่อฟิลด์ ต้องการทุกฟิลด์ ใช้ * ดอกจัน

คำสั่ง **FROM** จะตามด้วย ชื่อตาราง คือ employee

คำสั่ง WHERE จะตามด้วยเงื่อนไข โดยมี ชื่อฟิลด์ เปรียบเทียบ ค่า เช่นกรณีนี้ คือ emp_dept = 'บัญชี' AND emp_salary > 20000 หมายความว่า ต้องการแสดงข้อมูลพนักงานแผนกบัญชี ที่มีเงินเดือนมากกว่า 20000

คำสั่ง LIMIT จะเริ่มต้นที่ record ที่เท่าไร แล้วนำมากี่ record กรณีนี้ Limit 0,30 คือ ต้องการตั้งแต่ record แรก แล้วนำมาแสดง 30 record ถ้ามี

ตัวอย่างที่ 5-6 แสดงข้อมูลพนักงานแผนกบัญชี กับแผนกคอมพิวเตอร์

แสดงระเบียนที่ 0 - 1 (2 ทั้งหมด, ค่าค้นใช้เวลา 0.0006 วินาที)

คำสั่ง SQL:
 SELECT *
 FROM employee
 WHERE emp_dept = 'บัญชี'
 OR emp_dept = 'คอมพิวเตอร์' LIMIT 0 , 30

[แก้ไข] [อธิบาย SQL] [สร้างโค้ด PHP] [Refresh]

แสดง : แถว เริ่มจากแถวที่

อยู่ใน และซ้ำหัวแถวทุกๆ เซลล์

Sort by key:

	emp_id	emp_name	emp_surname	emp_dept	emp_salary
☐	5	สมชาย	แซ่เฮง	บัญชี	25000
☐	4	ดร.ณี	ชีชาลานนท์	คอมพิวเตอร์	8500

รูปที่ 5-15 : แบบสอบถามแสดงพนักงานแผนกบัญชีกับแผนกคอมพิวเตอร์

อธิบายรหัสคำสั่ง

คำสั่ง SELECT จะตามด้วย ชื่อฟิลด์ ต้องการทุกฟิลด์ ใช้ * ดอกจัน

คำสั่ง FROM จะตามด้วย ชื่อตาราง คือ employee

คำสั่ง WHERE จะตามด้วยเงื่อนไข โดยมี ชื่อฟิลด์ เปรียบเทียบ ค่า เช่นกรณีนี้ คือ emp_dept = 'บัญชี' OR emp_dept = 'คอมพิวเตอร์' หมายความว่า ต้องการแสดงข้อมูลพนักงานแผนกบัญชี กับพนักงานแผนกคอมพิวเตอร์

คำสั่ง LIMIT จะเริ่มต้นที่ record ที่เท่าไร แล้วนำมากี่ record กรณีนี้ Limit 0,30 คือ ต้องการตั้งแต่ record แรก แล้วนำมาแสดง 30 record ถ้ามี

ตัวอย่างที่ 5-7 แสดงข้อมูลพนักงาน ที่มีเงินเดือนระหว่าง 8500 ถึง 20,000 บาท

แสดงระเบียนที่ 0 - 0 (1 ทั้งหมด, ค่าค้นใช้เวลา 0.0005 วินาที)

คำสั่ง SQL:
 SELECT *
 FROM employee
 WHERE emp_salary
 BETWEEN 8500
 AND 20000 LIMIT 0 , 30

[แก้ไข] [อธิบาย SQL] [สร้างโค้ด PHP] [Refresh]

แสดง : 30 แถว เริ่มจากแถวที่ 0

อยู่ใน แนวนอน และซ้ำหัวแถวทุกๆ 100 เซลล์

←T→	emp_id	emp_name	emp_surname	emp_dept	emp_salary
 	4	ดร.สุณี	ชีชาลาณนท์	คอมพิวเตอร์	8500

รูปที่ 5-16 : แสดงการใช้ Between

อธิบายรหัสคำสั่ง

คำสั่ง SELECT จะตามด้วย ชื่อฟิลด์ ต้องการทุกฟิลด์ ใช้ * ดอกจัน

คำสั่ง FROM จะตามด้วย ชื่อตาราง คือ employee

คำสั่ง WHERE จะตามด้วยเงื่อนไข โดยมี ชื่อฟิลด์ เปรียบเทียบ ค่า เช่นกรณีนี้ emp_salary between 8500 AND 20000 หมายความว่า ต้องการแสดงข้อมูลพนักงาน ที่มีเงินเดือนตั้งแต่ 8500 ถึง 20000

คำสั่ง LIMIT จะเริ่มต้นที่ record ที่เท่าไร แล้วนำมากี่ record กรณีนี้ Limit 0,30 คือ ต้องการตั้งแต่ record แรก แล้วนำมาแสดง 30 record ถ้ามี

ตัวอย่างที่ 5-8 แสดงข้อมูลพนักงาน ที่มีเงินเดือนไม่เท่ากับ 8500 บาท

phpMyAdmin - Microsoft Internet Explorer

SQL นำเข้าไฟล์ SQL-history

ประมวลผลคำสั่งบนฐานข้อมูล oa

SELECT * FROM employee
where emp_salary <> 8500

จำนวนฟิลด์:
 emp_id
 emp_name
 emp_surname
 emp_dept
 emp_salary

☒ แสดงคำสั่งนี้อีกครั้ง
☒ Do not overwrite this query from outside the window

ลงมือ

Done Local intranet

รูปที่ 5-17 : แสดงการใช้ เครื่องหมายไม่เท่ากับกับตัวเลข

อธิบายรหัสคำสั่ง

คำสั่ง SELECT จะตามด้วย ชื่อฟิลด์ ต้องการทุกฟิลด์ ใช้ * ดอกจัน

คำสั่ง FROM จะตามด้วย ชื่อตาราง คือ employee

คำสั่ง WHERE จะตามด้วยเงื่อนไข โดยมี ชื่อฟิลด์ เปรียบเทียบ ค่า เช่น emp_salary

<> 8500 หมายความว่า ต้องการ แสดงข้อมูลพนักงานที่มีเงินเดือนไม่เท่ากับ 8500

คำสั่ง LIMIT จะเริ่มต้นที่ record ที่เท่าไร แล้วนำมากี่ record กรณีนี้ Limit 0,30 คือ ต้องการตั้งแต่ record แรก แล้วนำมาแสดง 30 record ถ้ามี

ตัวอย่างที่ 5-9 แสดงข้อมูลพนักงาน ที่มีไม่ได้อยู่แผนกคอมพิวเตอร์

แสดงระเบียนที่ 0 - 0 (1 ทั้งหมด, ค่าค้นใช้เวลา 0.0006 วินาที)

คำสั่ง SQL:
SELECT *
FROM employee
WHERE emp_dept <> 'คอมพิวเตอร์' **LIMIT** 0 , 30

[แก้ไข] [อธิบาย SQL] [สร้างโค้ด PHP] [Refresh]

แสดง : 30 แถว เริ่มจากแถวที่ 0

อยู่ใน และข้ามหัวแถวทุกๆ เซลล์

	emp_id	emp_name	emp_surname	emp_dept	emp_salary
☐	5	สมชาย	แซ่เซ่ง	บัญชี	25000

รูปที่ 5-18 : แสดงการใช้ เครื่องหมายไม่เท่ากับกับข้อความ

อธิบายรหัสคำสั่ง

คำสั่ง SELECT จะตามด้วย ชื่อฟิลด์ ต้องการทุกฟิลด์ ใช้ * ดอกจัน

คำสั่ง FROM จะตามด้วย ชื่อตาราง คือ employee

คำสั่ง WHERE จะตามด้วยเงื่อนไข โดยมี ชื่อฟิลด์ เปรียบเทียบ ค่า เช่น emp_dept

<> 'คอมพิวเตอร์' หมายความว่า ต้องการแสดงข้อมูลพนักงานที่ไม่ได้อยู่ในแผนก คอมพิวเตอร์

คำสั่ง LIMIT จะเริ่มต้นที่ record ที่เท่าไร แล้วนำมากี่ record กรณีนี้ Limit 0,30 คือ ต้องการตั้งแต่ record แรก แล้วนำมาแสดง 30 record ถ้ามี

ตัวอย่างที่ 5-10 แสดงข้อมูลพนักงาน ที่มีอยู่แผนกคอมพิวเตอร์

แสดงระเบียนที่ 0 - 0 (1 ทั้งหมด, ค่าค้นใช้เวลา 0.0005 วินาที)

คำสั่ง SQL:
SELECT *
FROM employee
WHERE emp_dept **LIKE** 'คอมพิวเตอร์' **LIMIT** 0 , 30

[แก้ไข] [อธิบาย SQL] [สร้างโค้ด PHP] [Refresh]

แสดง : 30 แถว เริ่มจากแถวที่ 0

อยู่ใน และซ้ำหัวแถวทุกๆ เซลล์

←T→	emp_id	emp_name	emp_surname	emp_dept	emp_salary
☐  	4	ดร.ณัฏฐ์	ชัชวาลานนท์	คอมพิวเตอร์	8500

รูปที่ 5-19 : แสดงการใช้ เครื่องหมาย LIKE

อธิบายรหัสคำสั่ง

คำสั่ง SELECT จะตามด้วย ชื่อฟิลด์ ต้องการทุกฟิลด์ ใช้ * ดอกจัน

คำสั่ง FROM จะตามด้วย ชื่อตาราง คือ employee

คำสั่ง WHERE จะตามด้วยเงื่อนไข โดยมี ชื่อฟิลด์ เปรียบเทียบ ค่า เช่น emp_dept like 'คอมพิวเตอร์' หมายความว่า ต้องการแสดงข้อมูลพนักงานที่อยู่แผนกคอมพิวเตอร์

คำสั่ง LIMIT จะเริ่มต้นที่ record ที่เท่าไร แล้วนำมากี่ record กรณีนี้ Limit 0,30 คือ ต้องการตั้งแต่ record แรก แล้วนำมาแสดง 30 record ถ้ามี

ตัวอย่างที่ 5-11 แสดงข้อมูลพนักงาน ที่ไม่มีอยู่แผนกคอมพิวเตอร์

แสดงระเบียนที่ 0 - 0 (1 ทั้งหมด, ค่าค้นใช้เวลา 0.0005 วินาที)

คำสั่ง SQL:
SELECT *
FROM employee
WHERE emp_dept **NOT LIKE** 'คอมพิวเตอร์' **LIMIT** 0 , 30

[แก้ไข] [อธิบาย SQL] [สร้างโค้ด PHP] [Refresh]

แสดง : 30 แถว เริ่มจากแถวที่ 0

อยู่ใน และซ้ำหัวแถวทุกๆ เซลล์

←T→	emp_id	emp_name	emp_surname	emp_dept	emp_salary
☐  	5	สมชาย	แซ่เฮง	บัญชี	25000

รูปที่ 5-20 : แสดงการใช้ เครื่องหมาย NOT LIKE

อธิบายรหัสคำสั่ง

คำสั่ง SELECT จะตามด้วย ชื่อฟิลด์ ต้องการทุกฟิลด์ ใช้ * ดอกจัน

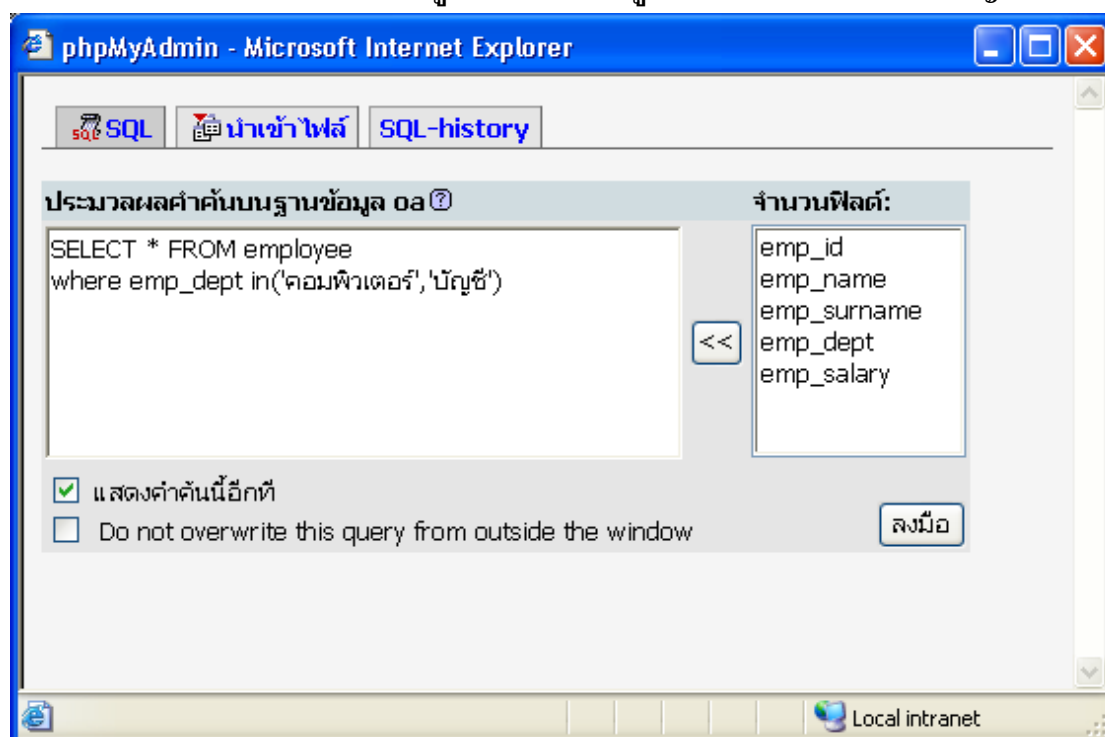
คำสั่ง FROM จะตามด้วย ชื่อตาราง คือ employee

คำสั่ง WHERE จะตามด้วยเงื่อนไข โดยมี ชื่อฟิลด์ เปรียบเทียบ ค่า เช่น emp_dept NOT LIKE 'คอมพิวเตอร์'

หมายความว่า ต้องการแสดงข้อมูลพนักงานที่ไม่ได้อยู่แผนกคอมพิวเตอร์

คำสั่ง LIMIT จะเริ่มต้นที่ record ที่เท่าไร แล้วนำมากี่ record กรณีนี้ Limit 0,30 คือ ต้องการตั้งแต่ record แรก แล้วนำมาแสดง 30 record ถ้ามี

ตัวอย่างที่ 5-12 แสดงข้อมูลพนักงาน ที่อยู่แผนกคอมพิวเตอร์ กับบัญชี



รูปที่ 5-21 : แสดงการใช้ IN

อธิบายรหัสคำสั่ง

คำสั่ง SELECT จะตามด้วย ชื่อฟิลด์ ต้องการทุกฟิลด์ ใช้ * ดอกจัน

คำสั่ง FROM จะตามด้วย ชื่อตาราง คือ employee

คำสั่ง WHERE จะตามด้วยเงื่อนไข โดยมี ชื่อฟิลด์ เปรียบเทียบ ค่า เช่น emp_dept in ('คอมพิวเตอร์','บัญชี') หมายความว่า ต้องการแสดงข้อมูลพนักงานที่อยู่ในแผนกคอมพิวเตอร์ กับแผนกบัญชี

ตัวอย่างที่ 5-13 แสดงข้อมูลโดยใช้ %

แสดงระเบียนที่ 0 - 0 (1 ทั้งหมด, ค่าค้นใช้เวลา 0.0005 วินาที)

คำสั่ง SQL:
SELECT *
FROM employee
WHERE emp_name **LIKE** '%ณั' **LIMIT** 0 , 30

[แก้ไข] [อธิบาย SQL] [สร้างโค้ด PHP] [Refresh]

แสดง : 30 แถว เริ่มจากแถวที่ 0

อยู่ใน แถวบน และซ้ำหัวแถวทุกๆ 100 เซลล์

←T→	emp_id	emp_name	emp_surname	emp_dept	emp_salary
☐	4	ดร.ณั	ชัชวาลานนท์	คอมพิวเตอร์	8500

รูปที่ 5-22 : แสดงการใช้ %

อธิบายรหัสคำสั่ง

คำสั่ง SELECT จะตามด้วย ชื่อฟิลด์ ต้องการทุกฟิลด์ ใช้ * ดอกจัน

คำสั่ง FROM จะตามด้วย ชื่อตาราง คือ employee

คำสั่ง WHERE จะตามด้วยเงื่อนไข โดยมี ชื่อฟิลด์ เปรียบเทียบ ค่า เช่น emp_name like ' % ณั ' หมายความว่า ต้องการ แสดง ข้อมูล พนักงาน ที่มี ชื่อ นำหน้าด้วยอะไรก็ได้แต่ต้องลงท้ายด้วย ณั

คำสั่ง LIMIT จะเริ่มต้นที่ record ที่เท่าไร แล้วนำมากี่ record กรณีนี้ Limit 0,30 คือ ต้องการตั้งแต่ record แรก แล้วนำมาแสดง 30 record ถ้ามี

ตัวอย่างที่ 5-14 แสดงข้อมูลโดยใช้ คำสั่ง ORDER BY

แสดงระเบียนที่ 0 - 1 (2 ทั้งหมด, ค่าค้นใช้เวลา 0.0006 วินาที)

คำสั่ง SQL:
SELECT *
FROM employee
ORDER BY emp_salary **DESC**
LIMIT 0 , 2

[แก้ไข] [อธิบาย SQL] [สร้างโค้ด PHP] [Refresh]

แสดง : 30 แถว เริ่มจากแถวที่ 0

อยู่ใน แถวบน และซ้ำหัวแถวทุกๆ 100 เซลล์

Sort by key: *ไม่มี ลงมือ

←T→	emp_id	emp_name	emp_surname	emp_dept	emp_salary
☐	5	สมชาย	แซ่เฮง	บัญชี	25000
☐	4	ดร.ณั	ชัชวาลานนท์	คอมพิวเตอร์	8500

รูปที่ 5-23 : แสดงการใช้ Order By

อธิบายรหัสคำสั่ง

คำสั่ง SELECT จะตามด้วย ชื่อฟิลด์ ต้องการทุกฟิลด์ ใช้ * ดอกจัน

คำสั่ง FROM จะตามด้วย ชื่อตาราง คือ employee

คำสั่ง ORDER BY จะตามด้วย ชื่อฟิลด์ ต้องการเรียง คือ emp_salary ปกติจะเรียงจากน้อยไปมาก ถ้าต้องการเรียงจากมากไปน้อยให้ใส่ DESC

ตัวอย่างที่ 5-15 แสดงข้อมูลโดยใช้ คำสั่ง INSERT

The screenshot shows a window titled "ประมวลผลคำสั่งบนฐานข้อมูล oa: ②". The main text area contains the SQL command: `insert into employee values( ' ', 'วิวัฒน์', 'จินนาท', 'บัญชี', 9500)`. At the bottom left, there is a checkbox labeled "แสดงคำค้นนี้อีกที" which is checked. At the bottom right, there is a button labeled "ลงมือ".

รูปที่ 5-24 : แสดงการใช้ คำสั่ง INSERT

อธิบายรหัสคำสั่ง

คำสั่ง Insert จะตามด้วย Into และ ชื่อตาราง คือ employee และ value แล้วใส่ค่าที่ต้องการ เพิ่มข้อมูลเข้าไป คือ รหัสพนักงานเป็น auto_number จึงไม่ต้องใส่ ใช้ " single quotes 2 ตัวติดกัน ตามด้วยชื่อ นามสกุล และชื่อแผนก อยู่ในเครื่องหมาย ' single quotes เพราะเป็นข้อความ และเงินเดือน ไม่ต้องใส่ " single quotes เพราะเป็นข้อมูลตัวเลข

ตัวอย่างที่ 5-16 แสดงข้อมูลโดยใช้ คำสั่ง UPDATE

The screenshot shows a window titled "ประมวลผลคำสั่งบนฐานข้อมูล oa: ②". The main text area contains the SQL command: `update employee set emp_salary = emp_salary*1.20`. On the right side, there is a panel titled "จำนวนฟิลด์:" containing a list of fields: emp_id, emp_name, emp_surname, emp_dept, and emp_salary. Below this list is a button labeled "<<". At the bottom left, there is a checkbox labeled "แสดงคำค้นนี้อีกที" which is checked. At the bottom right, there is a button labeled "ลงมือ".

รูปที่ 5-25 : แสดงการใช้ คำสั่ง Update

อธิบายรหัสคำสั่ง

คำสั่ง Update จะตามด้วย ชื่อตาราง คือ employee

โดยกำหนดว่าต้องการแก้ไขค่าในฟิลด์อะไรบ้าง เช่น ต้องการเพิ่มเงินเดือนจากเดิม 20 %

เขียนเป็น $\text{emp_salary} = \text{emp_salary} * 1.20$ โดยเงื่อนไข ไม่มี หมายความว่า

เพิ่มเงินเดือนให้กับพนักงานทุกคน

ตัวอย่างที่ 5-17 แสดงข้อมูลโดยใช้ คำสั่ง UPDATE แบบมีเงื่อนไข

The screenshot shows a database management interface. At the top, there's a title bar that says "ประมวลผลคำสั่งบนฐานข้อมูล oa: ๗". Below it is a text area containing the SQL command: `update employee set emp_salary=emp_salary*1.20 where emp_dept = 'คอมพิวเตอร์'`. Below the text area, there's a checkbox labeled "แสดงคำสั่งนี้อีกที" which is checked, and a button labeled "ลงมือ". Below this, there's a section titled "แถวที่มีผล: 1 (คำสั่งใช้เวลา 0.0005 วินาที)". Underneath, it says "คำสั่ง SQL:" followed by the same SQL command in a highlighted format: `UPDATE employee SET emp_salary = emp_salary * 1.20 WHERE emp_dept = 'คอมพิวเตอร์'`. At the bottom, there are two links: "[แก้ไข]" and "[สร้างโค้ด PHP]".

รูปที่ 5-26 : แสดงการใช้ คำสั่ง Update แบบมีเงื่อนไข

อธิบายรหัสคำสั่ง

คำสั่ง Update จะตามด้วย ชื่อตาราง คือ employee

โดยกำหนดว่าต้องการแก้ไขค่าในฟิลด์อะไรบ้าง เช่น ต้องการเพิ่มเงินเดือนจากเดิม 20 %

เขียนเป็น $\text{emp_salary} = \text{emp_salary} * 1.20$ โดยมีเงื่อนไข ในส่วนของ where

โดยกำหนดว่าจะเพิ่มเงินเดือนให้ใครบ้าง ในกรณีนี้ จะเพิ่มให้กับพนักงานแผนก คอมพิวเตอร์

จะใช้เงื่อนไข $\text{emp_dept} = \text{"คอมพิวเตอร์"}$

ตัวอย่างที่ 5-18 แสดงข้อมูลโดยใช้ คำสั่ง DELETE แบบมีเงื่อนไข

แถวที่ถูกลบ: 1 (ค่าคืนใช้เวลา 0.0005 วินาที)
 คำค้น SQL:
DELETE FROM employee WHERE emp_id = 7
 [แก้ไข] [สร้างโค้ด PHP]

รูปที่ 5-27 : แสดงการใช้ คำสั่ง DELETE

อธิบายรหัสคำสั่ง

คำสั่ง Delete จะตามด้วย ชื่อตาราง คือ employee โดยมีเงื่อนไข ในในส่วนของ where โดยกำหนดว่าจะลบ รหัสพนักงานคนใด เช่น emp_id = 7 คือต้องการลบข้อมูลจากตาราง employee ในระเบียนที่มี รหัสพนักงานเท่ากับ 7

แนวพระราชดำริเกี่ยวกับการศึกษา ของพระบาทสมเด็จพระเจ้าอยู่หัว

"...งานด้านการศึกษา เป็นงานสำคัญที่สุดอย่างหนึ่งของชาติ เพราะความเจริญและความเสื่อมของชาตินั้น ขึ้น อยู่กับการศึกษาของ พลเมืองเป็นข้อใหญ่ จึงต้องจัดการศึกษาให้เข้มแข็ง ยิ่งขึ้น..."

พระบรมราโชวาท ในพิธีพระราชทานปริญญาบัตร ณ วิทยาลัยวิชาการศึกษาประสานมิตร 12 ธันวาคม 2512

"...มหาวิทยาลัย มุ่งสั่งสอนนักศึกษาให้เป็นคนเก่ง ซึ่งเป็นการดี แต่นอกจากจะสอนให้เก่งแล้วจำเป็นอย่างยิ่ง ที่จะอบรมให้ดีพร้อมกันไปด้วย ประเทศเราจึงจะได้คนที่มีคุณภาพพร้อมคือ ทั้งเก่งและทั้งดีมาเป็นกำลัง ของบ้านเมือง..."

พระบรมราโชวาท พระราชทานแก่คณะผู้บริหาร และสภามหาวิทยาลัยต่าง ๆ

ณ ศาลาดุสิดาลัย 3 ตุลาคม 2533

"...การศึกษาเป็นเครื่องอันสำคัญในการพัฒนา ความรู้ ความคิด ความประพฤติ ทศนิยมและคุณธรรมของบุคคล เพื่อให้เป็น พลเมืองที่มีคุณภาพและประสิทธิภาพ เมื่อบ้านเมืองประกอบไปด้วย พลเมืองที่มีคุณภาพและประสิทธิภาพ การพัฒนาประเทศ ชาติก็ย่อม ทำให้ได้โดยสะดวกราบรื่นได้ผลที่แน่นอน และรวดเร็ว..."

พระราชดำรัส พระราชทานแก่ครูใหญ่ และนักเรียน ณ ศาลาดุสิดาลัย พระราชวังดุสิต

22 กรกฎาคม 2520

"...การสร้างสรรค์ความเจริญก้าวหน้าทุกอย่างนั้น ต้องเริ่มต้นที่ การศึกษาพื้นฐานเดิมก่อน เมื่อได้ศึกษาทราบ ชัดถึงส่วนดีส่วนเสียแล้ว จึง รักษาส่วนดีที่มีอยู่แล้วให้คงไว้ แล้วพยายาม ปรับปรุงสร้างเสริมด้วยหลัก วิชา อันประกอบด้วยเหตุผลและความสุจริตจริงใจ ให้ค่อยเจริญงอกงามมั่นคงบริบูรณ์ยิ่ง ๆ ขึ้นไป..."

พระบรมราโชวาท ในพิธีพระราชทานปริญญาบัตร ของ จุฬาลงกรณ์มหาวิทยาลัย 1 พฤศจิกายน 2528

"...ความรู้ที่ใช้ได้ผลนั้น ต้องเป็นความรู้ ที่ถูกต้อง แม่นยำ ชำนาญ นำมาใช้ในการได้ทันที และนอกจากความรู้ด้าน ลึก คือ วิชาเฉพาะสาขาที่ ศึกษาโดยตรงแล้ว ความรู้ด้านกว้าง คือ วิชาการอื่น ๆ ทั่วไป ย่อม เป็นปัจจัยประกอบส่งเสริมอีกส่วนหนึ่งด้วย..."

พระบรมราโชวาท ในพิธีพระราชทานปริญญาบัตร ของ มหาวิทยาลัยเชียงใหม่ 24 มกราคม 2532

"...การที่ได้มีโอกาสมาเรียนในต่างประเทศ นับว่าเป็นประโยชน์ เพราะหลายประเทศ มีความก้าวหน้าในด้านวิชาการ แต่ควรพิจารณา ด้วยสติปัญญาว่า อะไรที่ควรจะได้รับมาเป็น ประโยชน์แก่บ้านเมือง แล้วนำ เอาวิชาความรู้กลับไปช่วยบ้านเมือง..."

พระราชดำรัส พระราชทานแก่นักเรียนไทยในสหรัฐอเมริกา

ณ โรงแรมพลาซ่า นิวยอร์ก 8 มิถุนายน 2510

"...โลกปัจจุบันเต็มไปด้วยการโฆษณาชวนเชื่อ ฉะนั้น ก่อนที่จะปักใจเชื่ออะไรลงไปควรพิจารณาดู เหตุผลให้ต้องแท้เสียก่อน แม้แต่ สมเด็จพระสัมมาสัมพุทธเจ้า ยังทรงแนะนำให้ใช้สติ และปัญญา ศึกษาค้นคว้าและไตร่ตรองให้แน่ว่า คำสั่งสอน นั้นเป็นความจริงที่เชื่อได้หรือไม่ ไม่ให้สักแต่เชื่อเพราะว่ามีผู้รับญัตติไว้..."

พระราชดำรัส ในพิธีถวายปริญญาเกิตติมศักดิ์ ของ มหาวิทยาลัยวิลเลียมส์

ณ วิลเลียมทาวน์ นครนิวยอร์ก 11 มิถุนายน 2510

"...การพึ่งตนเองนั้นขึ้นอยู่กับคุณสมบัติประจำ บุคคลสองอย่าง คือความสามารถนำวิชาการที่ได้ศึกษามาใช้ ใน การปฏิบัติงาน กับความ ฉลาดที่จะวินิจฉัยให้เห็นทางเชื่อมทางเจริญพร้อม ทั้งทางที่จะให้พ้น ความเสื่อมเพื่อดำเนินไปให้ถึงความเจริญ..."

พระบรมราโชวาท ในพิธีพระราชทานปริญญาบัตร ของ มหาวิทยาลัยธรรมศาสตร์ 18 ธันวาคม 2512

"การที่มีการศึกษาสมบูรณ์แล้วนี้ ทำให้แต่ละคนหลีกเลี่ยงไม่ได้ จากความรับผิดชอบที่จะต้องใช้ความรู้ สติปัญญา ของ ตนให้มี ประโยชน์ และเป็น ความเจริญวัฒนา แก่บ้านเมือง และส่วนรวม"

(๑๒ กรกฎาคม ๒๕๑๖)

คัดลอกจาก กรมประชาสัมพันธ์

ที่มา :: <http://www.sudipan.net/phpBB2/viewtopic.php?p=10552>

บทที่ 6 การเขียนโปรแกรมติดต่อฐานข้อมูล

การเขียนโปรแกรมเพื่อติดต่อฐานข้อมูล ในฐานข้อมูล MySQL เป็นพื้นฐานในการสร้างระบบสารสนเทศต่าง ๆ โดยมีการทำงานพื้นฐานคือ การแสดงข้อมูล การเพิ่มข้อมูล การแก้ไขข้อมูล และ การลบข้อมูล การค้นหาข้อมูล และโดยต้องมีการสร้างข้อมูล MySQL ไว้ก่อนแล้ว

การแสดงข้อมูลจากฐานข้อมูล

การแสดงข้อมูลจากฐานข้อมูล มี 5 ขั้นตอน คือ

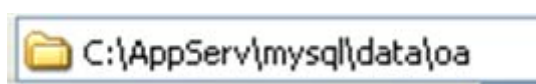
1. ติดต่อเครื่องแม่ข่าย
2. ติดต่อฐานข้อมูล
3. เขียนคำสั่ง SQL
4. รันคำสั่ง SQL
5. ดึงข้อมูลมาแสดงผล

การแสดงผลข้อมูล ต้องสร้างฐานข้อมูลในฐานข้อมูลเสียก่อน สามารถสร้างฐานข้อมูลได้โดยดูจากบทก่อนหน้านี้ ฐานข้อมูล ซึ่งสร้างฐานข้อมูล ชื่อ OA และตาราง ชื่อ EMPLOYEE มีจำนวนฟิลด์ อยู่ 5 ฟิลด์

	ฟิลด์	ชนิด	แอตทริบิวต์	ว่างเปล่า (null)	ค่าปริยาย	เพิ่มเติม	กระทำการ
☐	emp_id	int(11)		ไม่		auto_increment	     
☐	emp_name	varchar(20)		ไม่			     
☐	emp_surname	varchar(20)		ไม่			     
☐	emp_dept	varchar(15)		ไม่			     
☐	emp_salary	double		ไม่	0		     

รูปที่ 6-1 : แสดงตารางข้อมูลฟิลด์ในตารางพนักงาน

ซึ่งฐานข้อมูลที่สร้างไว้จะเก็บไว้ที่ C:\AppServ\mysql\data\ชื่อฐานข้อมูล ประกอบด้วยไฟล์ 3 นามสกุล คือ นามสกุล frm , MYD และ MYI



 employee.frm	9 KB	Visual Basic Form File
 employee.MYD	1 KB	MYD File
 employee.MYI	2 KB	MYI File

รูปที่ 6-2 : แสดงชื่อไฟล์ตารางพนักงาน

การติดต่อ SERVER

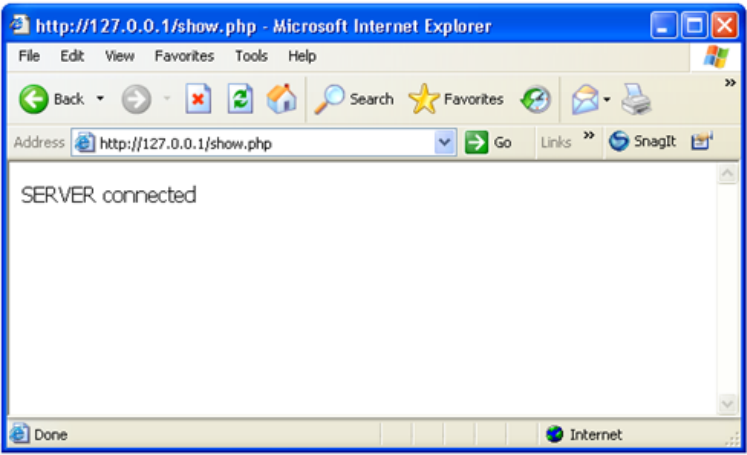
การติดต่อ SERVER จะใช้คำสั่ง `mysql_connect` โดยใส่พารามิเตอร์ให้ครบคือ เครื่องแม่ข่าย ใช้ `localhost` กรณีที่ ทดสอบที่เครื่องตนเอง ถ้าใช้จริงให้ใส่ ชื่อ Database Host ส่วนชื่อ ผู้ใช้ และรหัสผ่าน ผู้ให้บริการ ต้องมีชื่อ และ รหัสผ่านตามผู้ให้บริการได้จัดให้ สำหรับการทดสอบที่เครื่องตนเอง ใช้ user เป็น root และ password ไม่มี ให้ใส่ "" เครื่องหมายคำพูดติดกัน หรือบางเครื่องอาจกำหนดเป็น 1234 หรืออะไร ขึ้นอยู่ตอนติดตั้งโปรแกรม

ตัวอย่างที่ 6-1 ขั้นตอนการติดต่อเครื่องแม่ข่าย

```

1 <?
2 # ขั้นตอนที่ 1 ติดต่อ server
3 $conn = mysql_connect("localhost","root","") or die ("Cannot Connect
  SERVER");
4 echo "SERVER connected";
5 ?>

```



The screenshot shows a web browser window titled "http://127.0.0.1/show.php - Microsoft Internet Explorer". The address bar contains "http://127.0.0.1/show.php". The main content area displays the text "SERVER connected". The status bar at the bottom shows "Done" and "Internet".

รูปที่ 6-3 : แสดงการติดต่อเครื่องแม่ข่าย

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เครื่องหมาย <? หมายถึง เปิดคำสั่ง PHP

บรรทัดที่ 2 เครื่องหมาย # เป็นหมายเหตุ ของโปรแกรม PHP

บรรทัดที่ 3 เป็นคำสั่งติดต่อเครื่องแม่ข่ายโดยเครื่องแม่ข่ายชื่อ localhost ผู้ใช้ ชื่อ root ไม่มีรหัสผ่าน โดยหากไม่สามารถติดต่อได้ให้แสดงข้อความว่า “Cannot Connect SERVER”

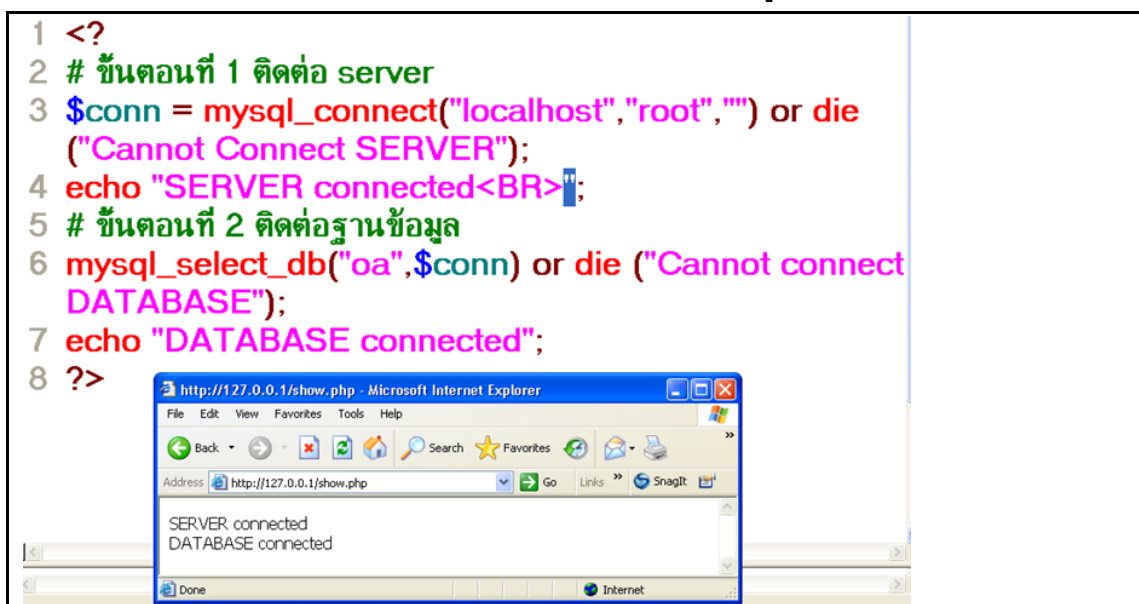
บรรทัดที่ 4 เป็นคำสั่งแสดงข้อความว่า “ SERVER connected “

หากโปรแกรมไม่สามารถติดต่อเครื่องแม่ข่ายได้ จะแสดงข้อความว่า Cannot Connect SERVER หากติดต่อได้ หมายความว่า ผ่านบรรทัดที่ 3 ได้ จะลงมาทำคำสั่ง บรรทัดที่ 4 ก็จะแสดงข้อความ SERVER connected แสดงว่า สามารถติดต่อเครื่องแม่ข่ายได้ ความผิดพลาดที่จะเกิดขึ้นในขั้นตอนนี้ จะเกิดกรณีที่ใส่ชื่อ เครื่องแม่ข่ายผิด ชื่อผู้ใช้ หรือรหัสผ่านผิด หรือเครื่องแม่ข่ายปิด

การติดต่อฐานข้อมูล

การติดต่อฐานข้อมูล จะใช้คำสั่ง mysql_select_db โดยใช้พารามิเตอร์ให้ครบคือ ชื่อฐานข้อมูล ใช้ localhost กรณีที่ ทดสอบที่เครื่องตนเอง ถ้าใช้จริงให้ใส่ ชื่อ Database Host ส่วนชื่อ ผู้ใช้ และรหัสผ่าน ผู้ให้บริการ ต้องมีชื่อ และ รหัสผ่านตามผู้ให้บริการได้จัดให้ สำหรับการทดสอบที่เครื่องตนเอง ใช้ user เป็น root และ password ไม่มี ให้ใส่ “” เครื่องหมายคำพูดติดกัน หรือบางเครื่องอาจกำหนดเป็น 1234 หรืออะไร ขึ้นอยู่ตอนติดตั้งโปรแกรม

ตัวอย่างที่ 6-2 การติดต่อฐานข้อมูล



รูปที่ 6-4 : แสดงการติดต่อฐานข้อมูล

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เครื่องหมาย <? หมายถึง เปิดคำสั่ง PHP

บรรทัดที่ 2 เครื่องหมาย # เป็น หมายเหตุ ของโปรแกรม PHP

บรรทัดที่ 3 เป็นคำสั่งติดต่อเครื่องแม่ข่ายโดยเครื่องแม่ข่ายชื่อ localhost ผู้ใช้ ชื่อ root ไม่มีรหัสผ่าน โดยหากไม่สามารถติดต่อได้ ให้แสดงข้อความว่า “Cannot Connect SERVER”

บรรทัดที่ 4 เป็นคำสั่งแสดงข้อความว่า “SERVER connected”

หากโปรแกรมไม่สามารถติดต่อเครื่องแม่ข่ายได้ จะแสดงข้อความว่า Cannot Connect SERVER หากติดต่อได้ หมายความว่า ผ่านบรรทัดที่ 3 ได้ จะลงมาทำคำสั่ง บรรทัดที่ 4 คือ จะแสดงข้อความ SERVER connected แสดงว่า สามารถติดต่อเครื่องแม่ข่ายได้ ความผิดพลาดที่เกิดขึ้นในขั้นตอนนี้ จะเกิดกรณีที่ใส่ชื่อ เครื่องแม่ข่ายผิด ชื่อผู้ใช้ หรือ รหัสผ่านผิด หรือเครื่องแม่ข่ายปิด

บรรทัดที่ 5 เครื่องหมาย # เป็น เหตุของโปรแกรม PHP

บรรทัดที่ 6 คำสั่ง mysql_select_db เป็นคำสั่งเพื่อติดต่อ ฐานข้อมูลชื่อ OA

ซึ่งอยู่ในเครื่องแม่ข่ายที่ติดต่อไว้แล้วตามตัวแปร conn

ซึ่งก็หมายถึงเครื่องแม่ข่ายที่ติดต่อไว้ในขั้นตอนที่ 1 นั่นเอง

โดยถ้าติดต่อฐานข้อมูลดังกล่าวไม่ได้จะแสดงข้อความ Cannot Connect DATABASE

บรรทัดที่ 7 เป็นคำสั่งแสดงข้อความว่า “Database connected”

โดยโปรแกรมจะแสดงข้อความนี้ได้ แสดงว่าผ่านขั้นตอนที่ 2 ได้แล้ว เพราะ

หากโปรแกรมไม่สามารถติดต่อฐานข้อมูลได้ จะแสดงข้อความว่า Cannot Connect DATABASE แล้วโปรแกรมจะไม่ทำงานต่อไป ความผิดพลาดที่เกิดขึ้นในขั้นตอนนี้ จะเกิดกรณีที่ใส่ชื่อ ฐานข้อมูลผิด

การรันคำสั่ง SQL

การรันคำสั่ง SQL เริ่มจากการเขียนคำสั่ง SQL เพื่อทำงานเช่นหากต้องการแสดงข้อมูล ใช้คำสั่ง Select From Where หากต้องการเพิ่มข้อมูล ก็ใช้คำสั่ง Insert Into หากต้องการแก้ไขข้อมูล ใช้คำสั่ง Update และถ้าต้องการลบข้อมูล ก็ใช้คำสั่ง Delete แล้วแต่กรณีและเงื่อนไขที่ต้องการ

ตัวอย่างที่ 6-3 การรันคำสั่ง SQL

```

8 # ขั้นตอนที่ 3 เขียนคำสั่ง SQL
9 $sql = "select * from employee ";
10 # ขั้นตอนที่ 4 รันคำสั่ง SQL
11 $result = mysql_query($sql) or die(mysql_error());
12 # ขั้นตอนที่ 5 ดึงข้อมูล
13 $rs = mysql_fetch_array($result);
14 echo $rs[1];
15 ?>

```

ผลลัพธ์

```

SERVER connected
DATABASE connected
สมชาย

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 8 เครื่องหมาย # เป็นหมายเหตุ ของโปรแกรม PHP

บรรทัดที่ 9 เป็นคำสั่ง SQL ที่จะแสดงข้อมูลทุกฟิลด์ในตาราง employee

บรรทัดที่ 10 เครื่องหมาย # เป็นหมายเหตุ ของโปรแกรม PHP

บรรทัดที่ 11 เป็นคำสั่งที่จะรันคำสั่ง SQL ที่เขียนไว้ในตัวแปร SQL

แล้วเก็บผลการรันไว้ในตัวแปร \$result หากมีข้อผิดพลาดให้แสดงข้อผิดพลาดออกมา

บรรทัดที่ 13 เป็นคำสั่งที่ดึง record ที่ละ record จากผลจะรันคำสั่ง SQL ที่เก็บไว้ใน ตัวแปร \$result แล้วนำ record นั้นเก็บไว้ในตัวแปร \$rs

บรรทัดที่ 14 เป็นคำสั่งแสดงข้อมูลที่ได้จากการดึง record โดยแสดงฟิลด์ ที่ 2 ออกมา คือ ชื่อพนักงานคนแรก ซึ่งผลที่ได้ คือ สมชาย

การดึงข้อมูลมาแสดง

การดึงข้อมูล ที่ได้จากการรันคำสั่ง SQL นั้นกรณีที่เป็นคำสั่ง Select From Where นั้น ข้อมูลจะอยู่ในลักษณะตารางซึ่งการรันคำสั่งแต่ละครั้งอาจได้มาหลายระเบียน จึงต้องดึงมาทีละระเบียนเพื่อแสดงผล โดยใช้คำสั่ง mysql_fetch_array

ตัวอย่างที่ 6-4 การดึงระเบียนมาแสดงผล

```

12 # ขั้นตอนที่ 5 ดึงข้อมูล
13 $rs = mysql_fetch_array($result);
14 echo $rs[0] . " " . $rs[1] . " " . $rs["emp_surname"]
   . "<BR>";
15 $rs = mysql_fetch_array($result);
16 echo $rs[0] . " " . $rs[1] . " " . $rs["emp_surname"];
17 ?>

```

ผลลัพธ์

```

SERVER connected
DATABASE connected
5 สมชาย แซ่เฮง
4 ดร.ณี ชัยกาลานนท์

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 14 เป็นคำสั่งแสดงข้อมูลที่ได้จากการดึง record โดยแสดงฟิลด์ ที่ 1, 2 และ ฟิลด์ emp_surname ออกมา โดย index จะเริ่มตั้งแต่ 0 , 1 ไปเรื่อย ๆ โดยการอ้างอิงสามารถอ้างด้วย index หรือ ชื่อฟิลด์ก็ได้

บรรทัดที่ 15 เป็นคำสั่งดึง record ต่อไป มาเก็บไว้ในตัวแปร \$rs

บรรทัดที่ 16 เป็นคำสั่งแสดงข้อมูลที่ได้จากการดึง record โดยแสดงฟิลด์ ที่ 1, 2 และ ฟิลด์ emp_surname ออกมา โดย index จะเริ่มตั้งแต่ 0 , 1 ไปเรื่อย ๆ โดยการอ้างอิงสามารถอ้างด้วย index หรือ ชื่อฟิลด์ก็ได้

การดึงข้อมูลมาแสดงแบบวนซ้ำ

ข้อมูลที่ได้จากการรันคำสั่ง Query ในแต่ละครั้ง เราไม่สามารถรู้ได้ว่า จะมีจำนวนกี่ระเบียน ดังนั้น ในการแสดงผล เราจึงใช้คำสั่ง While เพื่อวนลูปดึงระเบียนมาแสดงจนกว่าข้อมูลจะหมด

ตัวอย่างที่ 6-5 การดึงข้อมูลมาแสดงแบบวนลูป

```

12 # ขั้นตอนที่ 5 ดึงข้อมูล
13 while ($rs = mysql_fetch_array($result))
14     echo $rs[1] . " " . $rs["emp_surname"] . "<BR>";
15 ?>

```

ผลลัพธ์

```

SERVER connected
DATABASE connected
สมชาย แซ่เซ่ง
ดร.ณี ขั้ววานนท์
ศรีบุญ สมบูรณ์

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 13 เป็นคำสั่งดึง record มาเก็บไว้ในตัวแปร \$rs โดยแต่ละรอบดึงทีละ record จนกว่าจะหมด record

บรรทัดที่ 14 เป็นคำสั่งแสดงข้อมูล โดยแสดงฟิลด์ ชื่อ และ ฟิลด์ นามสกุล ออกมา

การแสดงผลข้อมูลเป็นตาราง

การแสดงผลข้อมูล สามารถใช้ตารางกำกับข้อมูล เพื่อความสวยงามของตาราง และข้อมูล โดยใช้หลักการการแสดงผลของตาราง โดยใช้คำสั่ง PHP ประกอบเนื่องด้วยต้องดึงข้อมูลจากฐานข้อมูล

ตัวอย่างที่ 6-6 การแสดงผลข้อมูลเป็นตาราง

```

10 # ขั้นตอนที่ 5 ดึงข้อมูล
11 echo "<TABLE border=1>";
12 while ($rs = mysql_fetch_array($result))
13 {
14     echo "<TR>";
15     echo "<TD>".$rs[0]."</TD>";
16     echo "<TD>".$rs[1]."</TD>";
17     echo "<TD>".$rs[2]."</TD>";
18     echo "<TD>".$rs["emp_dept"]."</TD>";
19     echo "<TD>".$rs["emp_salary"]."</TD>";
20     echo "</TR>";
21 }
22 echo "</TABLE>";
23 ?>

```

5	สมชาย แซ่เซ่ง	บัญชี	30000
4	ดร.ณี ขั้ววานนท์	คอมพิวเตอร์	12240
6	ศรีบุญ สมบูรณ์	การตลาด	2400

รูปที่ 6-5 : แสดงการแสดงผลเป็นตาราง

อธิบายรหัสคำสั่ง

บรรทัดที่ 11 เป็นคำสั่งสร้างตาราง โดยมีเส้นขอบเท่ากับ 1 เนื่องด้วยอยู่ในโปรแกรม php จึงอยู่ภายในคำสั่ง echo หรือ print

บรรทัดที่ 12 เป็นคำสั่งดึง record มาเก็บไว้ในตัวแปร \$rs โดยแต่ละรอบดึงทีละ record จนกว่าจะหมด record

บรรทัดที่ 15 เป็นคำสั่งแสดงข้อมูล โดยแสดงฟิลด์ รหัสพนักงาน ออกมา ในคอลัมน์ที่ 1

บรรทัดที่ 16 เป็นคำสั่งแสดงข้อมูล โดยแสดงฟิลด์ ชื่อพนักงาน ออกมา ในคอลัมน์ที่ 2

บรรทัดที่ 17 เป็นคำสั่งแสดงข้อมูล โดยแสดงฟิลด์ นามสกุล ออกมา ในคอลัมน์ที่ 3

บรรทัดที่ 18 เป็นคำสั่งแสดงข้อมูล โดยแสดงฟิลด์ ชื่อแผนก ออกมา ในคอลัมน์ที่ 4

บรรทัดที่ 19 เป็นคำสั่งแสดงข้อมูล โดยแสดงฟิลด์ เงินเดือน ออกมา ในคอลัมน์ที่ 5

การแสดงผลรูปแบบตัวเลข

การแสดงผลข้อมูลที่เป็นตัวเลข เพื่อความสวยงามของการแสดงผล และเข้าใจง่าย เนื่องจากตัวเลขมีเครื่องหมาย comma (,) ช่วยแบ่งตัวเลขให้สามารถอ่านเข้าใจง่ายขึ้น

ตัวอย่างที่ 6-7 การแสดงผลรูปแบบตัวเลข

```
19 echo "<TD>".number_format($rs[4],2)."</TD>";
20 echo "</TR>";
```

ผลลัพธ์

5	สมชาย	แข่ง	บัญชี	30,000.00
4	ดรณี	ข้าบาลานท์	คอมพิวเตอร์	12,240.00
6	ศรีญ	สมบูรณ์	การตลาด	2,400.00

อธิบายรหัสคำสั่ง

บรรทัดที่ 19 เป็นคำสั่งแสดงข้อมูล โดยแสดงฟิลด์ เงินเดือน ในรูปแบบมีเครื่องหมายแบ่งและมีทศนิยม

การจัดรูปแบบข้อมูล

การจัดรูปแบบข้อมูลตัวเลข โดยสากลจะจัดข้อมูลขีดขวา เพื่อการอ่าน และสวยงาม เพราะข้อมูลตัวเลข มีความสำคัญเชิงปริมาณด้วยดังนั้น เพื่อแสดงปริมาณให้ชัดเจนมากยิ่งขึ้น ควรจัดข้อมูลตัวเลขขีดขวา

ตัวอย่างที่ 6-8 การจัดรูปแบบข้อมูล

```
18 echo "<TD align=right>" .
    number_format($rs[4],2) . "</TD>";
19 echo "</TR>";
```

ผลลัพธ์

5	สมชาย แซ่เฮง	บัญชี	30,000.00
4	ดร.ณี ขั้ววาลานนท์	คอมพิวเตอร์	12,240.00
6	ศรัญ สมบูรณ์	การตลาด	2,400.00

อธิบายรหัสคำสั่ง

บรรทัดที่ 18 เป็นคำสั่งจัดรูปแบบ ให้ข้อมูลแสดงข้อมูลโดยแสดงข้อมูลขีดขวา

การใส่หัวข้อยตาราง

การใส่หัวข้อยตาราง เพื่อแสดงข้อมูลในฐานข้อมูลให้ชัดเจนว่าในแต่ละฟิลด์นั้นแสดงถึงข้อมูลอะไร มีหัวข้อย่างไร และเพื่อความสวยงาม การแสดงผล ดวรจัดกึ่งกลางและตัวหนา เพื่อให้ความสำคัญที่หัวข้อยตารางโดยใช้แท็ก TH

ตัวอย่างที่ 6-9 การใส่หัวข้อยตาราง

```
11 echo "<TABLE border=1>";
12 echo "<TR>";
13 echo "<TH>รหัส</TH>";
14 echo "<TH>ชื่อพนักงาน</TH>";
15 echo "<TH>แผนก</TH>";
16 echo "<TH>เงินเดือน</TH>";
17 echo "</TR>";
```

ผลลัพธ์

รหัส	ชื่อพนักงาน	แผนก	เงินเดือน
5	สมชาย แซ่เฮง	บัญชี	30,000.00
4	ดร.ณี ขั้ววาลานนท์	คอมพิวเตอร์	12,240.00
6	ศรัญ สมบูรณ์	การตลาด	2,400.00

อธิบายรหัสคำสั่ง

บรรทัดที่ 11 เป็นคำสั่งสร้างตาราง โดยมีเส้นขอบ=1

บรรทัดที่ 12 เป็นคำสั่งสร้างแถวตาราง

บรรทัดที่ 13 เป็นคำสั่งสร้างหัวข้อตาราง เป็น รหัส โดยเป็นตัวหนาและอยู่กึ่งกลาง

บรรทัดที่ 14 เป็นคำสั่งสร้างหัวข้อตาราง เป็น ชื่อพนักงาน โดยเป็นตัวหนาและอยู่กึ่งกลาง

บรรทัดที่ 15 เป็นคำสั่งสร้างหัวข้อตาราง เป็น แผนก โดยเป็นตัวหนาและอยู่กึ่งกลาง

บรรทัดที่ 16 เป็นคำสั่งสร้างหัวข้อตาราง เป็น เงินเดือน โดยเป็นตัวหนาและอยู่กึ่งกลาง

บรรทัดที่ 17 เป็นคำสั่งปิดแถวตาราง

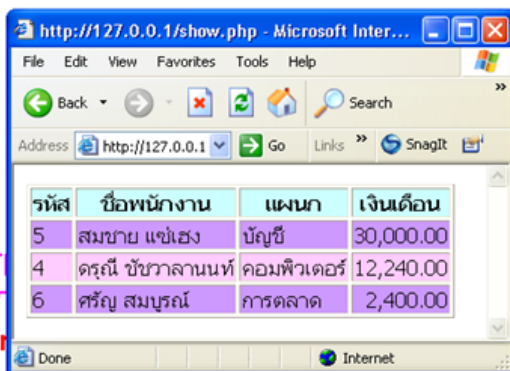
การแสดงผลสลับสี

การแสดงผลข้อมูล หากแสดงผลมุลปริมาณมาก ๆ ในการแสดงผลควรสลับสีในการแสดงผล เพื่อให้เห็นข้อมูลชัดเจนไม่ล้นตาเกินไป และเพื่อความสวยงามในการแสดงผล จึงควรแสดงข้อมูลสลับสี โดยใช้ตัวแปร และการเขียนโปรแกรมเงื่อนไข เข้ามาช่วยในการใส่สีพื้นของตาราง

ตัวอย่างที่ 6-10 การสลับสีตาราง

```

18 $c="#FFCCFF";
19 while ($rs = mysql_fetch_array($result))
20 {
21     if ($c == "#FFCCFF")
22         $c = "#CC99FF";
23     else
24         $c = "#FFCCFF";
25     echo "<TR bgcolor=$c>";
26     echo "<TD>".$rs[0]."</TD>";
27     echo "<TD>".$rs[1].".$rs[2]."</TD>";
28     echo "<TD>".$rs['emp_dept']."</TD>";
29     echo "<TD align=right>". number_format($rs[3], 2). "</TD>";
30     echo "</TR>";
31 }
```



รูปที่ 6-6 : แสดงการแสดงผลเป็นตารางแบบสลับสี

อธิบายรหัสคำสั่ง

บรรทัดที่ 18 เป็นคำสั่งตั้งค่าตัวแปร \$c เป็นสี ชมพู่

บรรทัดที่ 21-24 เป็นคำสั่งที่ตรวจสอบว่าเป็นสีอะไร ถ้าเป็นสีชมพู ก็เปลี่ยนเป็นสีม่วง หากเป็นสีม่วง จะเปลี่ยนเป็นสีชมพู

บรรทัดที่ 25 เป็นคำสั่งใส่สีพื้นของตารางเป็นสี ที่อยู่ในตัวแปร \$c คือสีชมพู หรือ สีม่วง

การเพิ่มข้อมูลในฐานข้อมูล

การเพิ่มข้อมูลลงฐานข้อมูล เริ่มจากฟอร์ม เพื่อรับข้อมูล เพื่อเพิ่มข้อมูล แล้วส่งให้ไฟล์ PHP เพื่อทำการติดต่อฐานข้อมูลแล้วเพิ่มข้อมูลลงฐานข้อมูล ซึ่งมี 5 ขั้นตอน คือ

1. ติดต่อเครื่องแม่ข่าย
2. ติดต่อฐานข้อมูล
3. เขียนคำสั่ง SQL
4. รันคำสั่ง SQL
5. ดึงข้อมูลมาแสดงผล

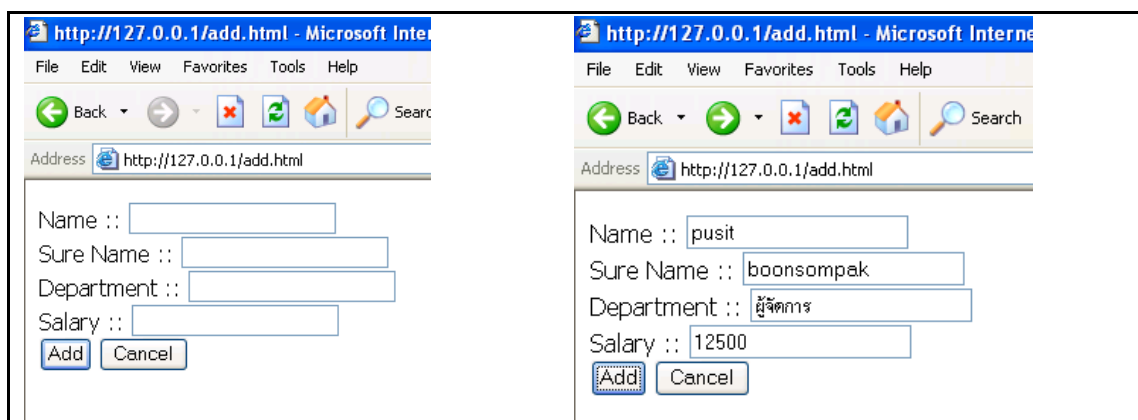
ตัวอย่างที่ 6-11 ไฟล์ add.html

```

1  <FORM METHOD=POST ACTION="add.php">
2  Name :: <INPUT TYPE="text" NAME="txtName"><BR>
3  Sure      Name      ::      <INPUT      TYPE="text"
NAME="txtSureName"><BR>
4  Department :: <INPUT TYPE="text" NAME="txtDept"><BR>
5  Salary :: <INPUT TYPE="text" NAME="txtSalary"><BR>
6  <INPUT TYPE="submit" value = "Add">
7  <INPUT TYPE="reset" value = "Cancel">
8  </FORM>

```

ผลลัพธ์



รูปที่ 6-7 : แสดงการคีย์ข้อมูลเพื่อเพิ่มข้อมูล

เมื่อมีการใส่ข้อมูล ข้อมูลจะถูกส่งไปยังฐานข้อมูล ที่ได้สร้างไว้

แสดง : 30 แถว เริ่มจากแถวที่ 0

อยู่ใน แนวนอน และเข้าหัวแถวทุกๆ 100 เซลล์

Sort by key: ไม่มี ลงมือ

	emp_id	emp_name	emp_surname	emp_dept	emp_salary	pic
☐	12	pusit	boonsompak	ผู้จัดการ	12500	
☐	4	dddd	ddd	ddddddddd	5000	logo.jpg
☐	5	5	5	5	5	
☐	6	88	88	88	88	
☐	7	99	99	99	99	

เลือกทั้งหมด / ไม่เลือกเลย ทำกับที่เลือก:

รูปที่ 6-8 : แสดงการเพิ่มข้อมูลในตาราง

อธิบายรหัสคำสั่ง

- บรรทัดที่ 1 เป็นการสร้างฟอร์ม เพื่อกำหนดไฟล์ทำงานต่อไป คือ add.php
- บรรทัดที่ 2 เป็นการแสดงข้อความ Name และกำหนดรับข้อมูลด้วย Text Box
- บรรทัดที่ 3 เป็นการแสดงข้อความ Sur Name และกำหนดรับข้อมูลด้วย Text Box
- บรรทัดที่ 4 เป็นการแสดงข้อความ Department และกำหนดรับข้อมูลด้วย Text Box
- บรรทัดที่ 5 เป็นการแสดงข้อความ Salary และกำหนดรับข้อมูลด้วย Text Box
- บรรทัดที่ 6 เป็นการกำหนดปุ่ม submit มีข้อความบนปุ่ม คือ Add
- บรรทัดที่ 7 เป็นการกำหนดปุ่ม reset มีข้อความบนปุ่ม คือ Cancel
- บรรทัดที่ 8 เป็นการปิดขอบเขตของฟอร์ม

ตัวอย่างที่ 6-12 ไฟล์ add.php

```

1 <?
2 $conn=mysql_connect("localhost","root","") or die("Cannot connect Server");
3 mysql_select_db("work1",$conn) or die("Cannot connect database");
4 $sql = "insert into employee(emp_id,emp_name,emp_surname,emp_dept,emp_salary)
values('$txtName','$txtSureName','$txtDept','$txtSalary')";
5 // echo $sql;
6 $result = mysql_query($sql) or die(mysql_error());
7 if ($result)
8     echo "insert to database 1 record<BR>";
9 else
10     echo "Cannot insert data<BR>";
11 mysql_close($conn) ?>
12 <A HREF="Display.php">Home</A>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นการเริ่มต้นเอกสาร PHP

บรรทัดที่ 2 เป็นการติดต่อเครื่องแม่ข่าย

บรรทัดที่ 3 เป็นการติดต่อฐานข้อมูล

บรรทัดที่ 4 เป็นคำสั่ง insert เพื่อเพิ่มข้อมูลลงฐานข้อมูล

บรรทัดที่ 5 เป็นการแสดงคำสั่ง SQL แต่ตีความเหตุไว้

บรรทัดที่ 6 เป็นการรันคำสั่ง SQL แล้วเก็บผลลัพธ์ไว้ที่ตัวแปร result

บรรทัดที่ 7-10 เป็นการตรวจสอบว่า สามารถเพิ่มข้อมูลได้หรือไม่ ถ้าได้ แสดงข้อความว่า

Insert to database 1 record ถ้าไม่ได้ แสดงข้อความว่า Cannot insert data

บรรทัดที่ 11 เป็นการปิดการเชื่อมต่อเครื่องแม่ข่าย ที่ได้เปิดไว้

บรรทัดที่ 12 เป็นการแสดง link เพื่อกลับไปไฟล์ Display.php เพื่อแสดงข้อมูลในฐานข้อมูลว่าข้อมูลได้เพิ่มในฐานข้อมูลถูกต้อง

การแก้ไขข้อมูล

การแก้ไขข้อมูลในฐานข้อมูล เริ่มจากหน้า display.php แสดงข้อมูลทั้งหมดของพนักงาน เมื่อคลิกเลือก record ที่ต้องการแก้ไขโดยจะส่ง รหัสพนักงานไปกับ link ด้วย โดย ฟังก์ชันของการ แก้ไข จะคล้ายกับ เพิ่มข้อมูล แต่ต้องมีข้อมูลจากฐานข้อมูลที่ต้องการแก้ไข ฉะนั้น ไฟล์ ฟังก์ชัน เพื่อรับการแก้ไขจึงต้องเป็นไฟล์ของ PHP เพราะต้องดึงข้อมูล จากฐานข้อมูล มาแสดง แล้วรอรับการแก้ไข เมื่อแก้ไขข้อมูลเสร็จจะส่งให้ไฟล์ PHP ประมวลผล เพื่อทำการ ติดต่อ ฐานข้อมูลแล้วแก้ไขข้อมูลในฐานข้อมูล ซึ่งมี 5 ขั้นตอน คือ

1. ติดต่อเครื่องแม่ข่าย
2. ติดต่อฐานข้อมูล
3. เขียนคำสั่ง SQL
4. รันคำสั่ง SQL
5. ดึงข้อมูลมาแสดงผล

ตัวอย่างที่ 6-13 ไฟล์ Display.php

```

1  <?
2  $conn=mysql_connect("localhost","root","") or die("Cannot connect Server");
3  mysql_select_db("work1",$conn) or die("Cannot connect database");
4  $sql = "select * from employee";
5  $result = mysql_query($sql) or die(mysql_error());
6  ?>
7  <TABLE border=2>
8  <? while ($rs = mysql_fetch_array($result)) { ?>
9  <TR>
10     <TD><? echo $rs[0] ?></TD>
11     <TD><? echo $rs[1] ?></TD>
12     <TD><? echo $rs[2] ?></TD>
13     <TD><? echo $rs[3] ?></TD>
14     <TD><? echo $rs["emp_salary"] ?></TD>
15     <TD><A HREF="edit.php?id=<?=$rs[0]?>">Edit</A></TD>
16     <TD><A HREF="delete.php?id=<?=$rs[0]?>">Delete</A></TD>
17 </TR>
18 <? } ?>
19 </TABLE>
20 <? mysql_close($conn) ?>
21 <A HREF="Add.html">Add</A>

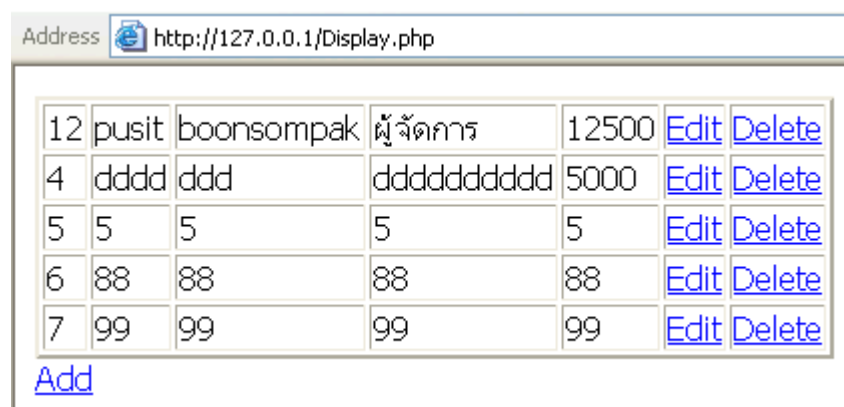
```

อธิบายรหัสคำสั่ง

บรรทัดที่ 15 เป็นการแสดง link เพื่อส่งข้อมูลไปแก้ไข โดยส่งรหัสพนักงานไปกับ link ด้วย

บรรทัดที่ 16 เป็นการแสดง link เพื่อส่งข้อมูลไปลบ โดยส่งรหัสพนักงานไปกับ link ด้วย

ผลลัพธ์



รูปที่ 6-9 : การแสดงผลพร้อมการเชื่อมต่อ

ในหน้า Display.php เราสามารถที่จะ Edit, Delete ข้อมูลที่มีอยู่ในฐานข้อมูลได้ รวมไปถึงยังสามารถ Add ข้อมูลใหม่ได้อีกด้วย

ตัวอย่างที่ 6-14 ไฟล์ Edit.php

```

1 <?
2 $conn=mysql_connect("localhost","root","") or die("Cannot connect Server");
3 mysql_select_db("work1",$conn) or die("Cannot connect database");
4 $sql = "select * from employee where emp_id = '$id' ";
5 $result = mysql_query($sql) or die(mysql_error());
6 $rs = mysql_fetch_array($result); ?>
7 <FORM METHOD=POST ACTION="update.php">
8 Id :: <INPUT TYPE="text" NAME="txtId" value="<?=$rs[0]?>" readonly ><BR>
9 Name :: <INPUT TYPE="text" NAME="txtName" value = "<?=$rs[1]?>" readonly><BR>
10 SureName :: <INPUT TYPE="text" NAME="txtSureName" value = "<?=$rs[2]?>"><BR>
11 Department :: <INPUT TYPE="text" NAME="txtDept" value = "<?=$rs[3]?>"><BR>
12 Salary :: <INPUT TYPE="text" NAME="txtSalary" value = "<?=$rs[4]?>"><BR>
13 <INPUT TYPE="submit" value = "Update">
14 <INPUT TYPE="reset" value = "Cancel">
15 </FORM>
16 <? mysql_Close($conn) ?>

```

ผลลัพธ์

Address http://127.0.0.1/edit.php

Id ::

Name ::

SureName ::

Department ::

Salary ::

รูปที่ 6-10 : แสดงการแก้ไขข้อมูลในตาราง

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นการเริ่มต้นเอกสาร PHP

บรรทัดที่ 2 เป็นการติดต่อเครื่องแม่ข่าย

บรรทัดที่ 3 เป็นการติดต่อฐานข้อมูล

บรรทัดที่ 4 เป็นคำสั่ง SQL เพื่อดึงข้อมูลในฐานข้อมูล โดยอ้างอิงตาม รหัสพนักงาน ที่ส่งมาพร้อมกับ link

บรรทัดที่ 5 เป็นการรันคำสั่ง SQL แล้วเก็บผลลัพธ์ไว้ที่ตัวแปร result

บรรทัดที่ 6 เป็นการดึง Record ที่ต้องการแก้ไข มาเก็บไว้ที่ ตัวแปร rs

บรรทัดที่ 7 เป็นการสร้างฟอร์ม เพื่อกำหนดไฟล์ทำงานต่อไป คือ update.php

บรรทัดที่ 8 เป็นการแสดงข้อความ Id และกำหนดรับข้อมูลด้วย Text Box โดยมีค่ารหัสพนักงานแสดงในช่อง และไม่สามารถแก้ไขได้

บรรทัดที่ 9 เป็นการแสดงข้อความ Name และกำหนดรับข้อมูลด้วย Text Box โดยมีค่าชื่อพนักงานแสดงในช่อง สามารถแก้ไขได้

บรรทัดที่ 10 เป็นการแสดงข้อความ Sur Name และกำหนดรับข้อมูลด้วย Text Box โดยมีค่าที่ดึงจากฐานข้อมูลมาแสดงในช่อง สามารถแก้ไขได้

บรรทัดที่ 11 เป็นการแสดงข้อความ Department และกำหนดรับข้อมูลด้วย Text Box โดยมีค่าที่ดึงจากฐานข้อมูลมาแสดงในช่อง สามารถแก้ไขได้

บรรทัดที่ 12 เป็นการแสดงข้อความ Salary และกำหนดรับข้อมูลด้วย Text Box โดยมีค่าที่ดึงจากฐานข้อมูลมาแสดงในช่อง สามารถแก้ไขได้

บรรทัดที่ 13 เป็นการกำหนดปุ่ม submit มีข้อความบนปุ่ม คือ Add

บรรทัดที่ 14 เป็นการกำหนดปุ่ม reset มีข้อความบนปุ่ม คือ Cancel

บรรทัดที่ 15 เป็นการปิดขอบเขตของฟอร์ม

บรรทัดที่ 16 เป็นการปิดการเชื่อมต่อเครื่องแม่ข่าย ที่ได้เปิดไว้

ตัวอย่างที่ 6-15 ไฟล์ Update.php

```

1 <?
2 $conn=mysql_connect("localhost","root","") or die("Cannot connect Server");
3 mysql_select_db("work1",$conn) or die("Cannot connect database");
4 $sql = "update employee set emp_name = '$txtName',emp_surname =
'$txtSureName',emp_dept='$txtDept',emp_salary=$txtSalary where emp_id = '$txtid' ";
5 echo $sql;
6 $result = mysql_query($sql) or die(mysql_error());
7 if ($result)
8     echo "Update to database 1 record<BR>";
9 else
10     echo "Cannot Update data<BR>";
11 mysql_close($conn) ?>
12 <A HREF="Display.php">Home</A>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นการเริ่มต้นเอกสาร PHP

บรรทัดที่ 2 เป็นการติดต่อเครื่องแม่ข่าย

บรรทัดที่ 3 เป็นการติดต่อฐานข้อมูล

บรรทัดที่ 4 เป็นคำสั่ง update เพื่อแก้ไขข้อมูลในฐานข้อมูล โดยอ้างอิงตาม รหัสพนักงานที่ส่งมาจากฟอร์ม edit.php

บรรทัดที่ 5 เป็นการแสดงคำสั่ง SQL เพื่อตรวจสอบการทำงาน

บรรทัดที่ 6 เป็นการรันคำสั่ง SQL แล้วเก็บผลลัพธ์ไว้ที่ตัวแปร result

บรรทัดที่ 7-10 เป็นการตรวจสอบว่า สามารถแก้ไขข้อมูลได้หรือไม่ ถ้าได้ แสดงข้อความว่า Update to database 1 record ถ้าไม่ได้ แสดงข้อความว่า Cannot UPDATE data
 บรรทัดที่ 11 เป็นการปิดการเชื่อมต่อเครื่องแม่ข่าย ที่ได้เปิดไว้
 บรรทัดที่ 12 เป็นการแสดง link เพื่อกลับไปไฟล์ Display.php เพื่อแสดงข้อมูลในฐานข้อมูลว่าข้อมูลได้แก้ไขข้อมูลในฐานข้อมูลถูกต้อง

การลบข้อมูลในฐานข้อมูล

การลบข้อมูลในฐานข้อมูล เริ่มจากหน้า display.php ที่แสดงข้อมูลทั้งหมดของพนักงาน เมื่อคลิกเลือก record ที่ต้องการลบโดยจะส่ง รหัสพนักงานไปกับ link ด้วยโดยไฟล์ PHP ประมวลผล เพื่อทำการติดต่อฐานข้อมูลแล้วลบข้อมูลในฐานข้อมูล ซึ่งมี 5 ขั้นตอน คือ

1. ติดต่อเครื่องแม่ข่าย
2. ติดต่อฐานข้อมูล
3. เขียนคำสั่ง SQL
4. รันคำสั่ง SQL
5. ดึงข้อมูลมาแสดงผล

ตัวอย่างที่ 6-16 ไฟล์ Delete.php

```

1 <?
2 $conn=mysql_connect("localhost","root","") or die("Cannot connect Server");
3 mysql_select_db("work1",$conn) or die("Cannot connect database");
4 $sql="delete from employee where emp_id='$id' ";
5 //echo $sql."<hr>";
6 $result = mysql_query($sql) or die(mysql_error());
7 if ($result)
8     echo "delete from database 1 record<BR>";
9 else
10    echo "Cannot delete data<BR>";
11 mysql_Close($conn) ?>
12 <A HREF="Display.php"><BR>Home</A>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นการเริ่มต้นเอกสาร PHP

บรรทัดที่ 2 เป็นการติดต่อเครื่องแม่ข่าย

บรรทัดที่ 3 เป็นการติดต่อฐานข้อมูล

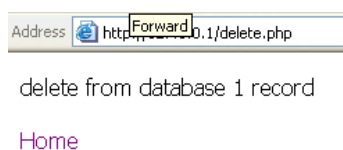
บรรทัดที่ 4 เป็นคำสั่ง delete เพื่อลบข้อมูลจากฐานข้อมูล โดยอ้างอิงตามรหัสพนักงานที่ส่งมาพร้อม link

บรรทัดที่ 5 เป็นการแสดงคำสั่ง SQL แต่ติดหมายเหตุไว้

บรรทัดที่ 6 เป็นการรันคำสั่ง SQL แล้วเก็บผลลัพธ์ไว้ที่ตัวแปร result

บรรทัดที่ 7-10 เป็นการตรวจสอบว่า สามารถลบข้อมูลได้หรือไม่ ถ้าได้ แสดงข้อความว่า Delete from database 1 record ถ้าไม่ได้ แสดงข้อความว่า Cannot Delete data
 บรรทัดที่ 11 เป็นการปิดการเชื่อมต่อเครื่องแม่ข่าย ที่ได้เปิดไว้
 บรรทัดที่ 12 เป็นการแสดง link เพื่อกลับไปไฟล์ Display.php เพื่อแสดงข้อมูลในฐานข้อมูลว่าข้อมูลได้ลบออกจากฐานข้อมูลถูกต้อง

ผลลัพธ์



รูปที่ 6-11 : แสดงผลลัพธ์การลบข้อมูล

แนวพระราชดำรัสเกี่ยวกับการศึกษา ของพระบาทสมเด็จพระเจ้าอยู่หัว

"ความรู้กับดวงประทีปเปรียบเทียบได้หลายทาง ดวงประทีปเป็นไฟที่ส่องแสงเพื่อนำทางไป ถ้าใช้ไฟนี้ส่องทางไปทางที่ถูกต้อง ก็จะไปถึงจุดหมายปลายทางได้โดยสะดวกเรียบร้อย และถ้าไม่ระวังไฟนั้น อาจจะเผาผลาญให้บ้านช่องพินาศลงได้ ความรู้เป็นแสงสว่างที่จะนำเราไปสู่ความเจริญ ถ้าไม่ระมัดระวังในการใช้ความรู้ก็จะเป็นอันตรายเช่นเดียวกัน จะทำลายเผาผลาญบ้านเมืองให้ล่มจมได้"

(๒๘ มกราคม ๒๕๐๕)

บทที่ 7 การใช้ตัวแปร Session

การใช้ตัวแปร Session เพื่อกำกับตัวแปรที่สำคัญ และจำเป็นต้องใช้ ข้ามหน้าเพจ หรือต้องการตรวจสอบการทำงานบางอย่าง เช่น การเข้าระบบ การตรวจสอบสิทธิการทำงาน หรือ ขั้นตอนการทำงาน โดยมีขั้นตอนการทำงาน คือ เปิดการทำงานของ Session ลงทะเบียนตัวแปร และทำลายตัวแปรเมื่อเลิกใช้

การลงทะเบียนตัวแปร Session

การลงทะเบียนตัวแปร Session เพื่อเป็นการเก็บตัวแปร Session ไว้ให้เพจอื่นสามารถใช้ตัวแปรนี้ได้ จนกว่าจะมีการทำลายตัวแปรดังกล่าวตัวอย่าง เช่น

ตัวอย่างที่ 7-1 โปรแกรม 1.php

```
1 <?
2 session_start();
3 $x=123;
4 session_register("x");
5 echo $_SESSION['x'];
6 session_register("y");
7 $_SESSION['y']=456;
8 echo $_SESSION['y']; ?>
9 <A href="2.php">next</A>
```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เครื่องหมาย <? หมายถึง เปิดคำสั่ง PHP

บรรทัดที่ 2 คำสั่ง session_start() เป็นคำสั่งเพื่อเปิดการทำงานใช้ตัวแปร Session

บรรทัดที่ 3 เป็นคำสั่งเพื่อกำหนดค่าให้กับตัวแปร \$x = 123

บรรทัดที่ 4 คำสั่ง session_register() เป็นคำสั่งเพื่อลงทะเบียนตัวแปร \$x ให้เป็นตัวแปร Session

บรรทัดที่ 5 เป็นคำสั่งเพื่อแสดงค่าตัวแปร \$x = 123 ซึ่งเป็นตัวแปร Session จึงต้องอ้างด้วย \$_SESSION

บรรทัดที่ 6 คำสั่ง session_register() เป็นคำสั่งเพื่อลงทะเบียนตัวแปร Session ชื่อ y

บรรทัดที่ 7 เป็นคำสั่งเพื่อกำหนดค่าให้กับตัวแปร Session y ให้เท่ากับ 456

บรรทัดที่ 8 เป็นคำสั่งเพื่อแสดงค่าตัวแปร \$y ซึ่งมีค่า 123 ซึ่งเป็นตัวแปร Session จึงต้องอ้างด้วย \$_SESSION

บรรทัดที่ 9 เป็นคำสั่งเพื่อสร้าง link ไปที่ไฟล์ 2.php

ผลลัพธ์

123456[next](#)

อธิบายผลลัพธ์

ผลลัพธ์ที่ได้ 123 คือค่าของตัวแปร Session x และ 456 คือค่าตัวแปร Session y

การทำลายตัวแปร Session

การทำลายตัวแปร Session จะใช้กรณีที่ไม่ต้องใช้ตัวแปร Session แล้ว จะใช้กรณีที่ออกจากระบบ ตัวแปร Session จะโดยทำลายได้หลายวิธี เช่น การปิด Browser, หมดอายุ, ใช้คำสั่ง `session_unregister()` หรือ `session_destroy()`

ตัวอย่างที่ 7-2 โปรแกรม 2.php

```
1 <?
2 session_start();
3 print $_SESSION['x'];
4 print $_SESSION['y'];
5 session_unregister('x');
6 session_destroy();
7 ?>
▶8 <A href="1.php">Back</A>
```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เครื่องหมาย <? หมายถึง เปิดคำสั่ง PHP

บรรทัดที่ 2 คำสั่ง `session_start()` เป็นคำสั่งเพื่อเปิดการทำงานใช้ตัวแปร Session

บรรทัดที่ 3 เป็นคำสั่งเพื่อแสดงค่าตัวแปร Session x = 123 ซึ่งเป็นตัวแปร Session จึงต้องอ้างด้วย `$_SESSION` ซึ่ง ตัวแปร Session x ต้องมีการลงทะเบียนแล้วจึงสามารถแสดงค่าได้

บรรทัดที่ 4 เป็นคำสั่งเพื่อแสดงค่าตัวแปร Session y = 456 ซึ่งเป็นตัวแปร Session จึงต้องอ้างด้วย `$_SESSION` ซึ่ง ตัวแปร Session y ต้องมีการลงทะเบียนแล้วจึงสามารถแสดงค่าได้

บรรทัดที่ 5 เป็นคำสั่งยกเลิกตัวแปร Session x เมื่อยกเลิกแล้วจะไม่สามารถอ้างถึงได้อีก

บรรทัดที่ 6 เป็นคำสั่งทำลายตัวแปร Session เมื่อทำลายแล้วจะไม่สามารถอ้างถึงได้อีก

บรรทัดที่ 7 เครื่องหมาย ?> หมายถึง ปิดคำสั่ง PHP

บรรทัดที่ 8 เป็นคำสั่งเพื่อสร้าง link ไปที่ไฟล์ 2.php

ผลลัพธ์ เมื่อรันต่อจากโปรแกรม 1.php

123456[next](#)

อธิบายผลลัพธ์

ผลลัพธ์ที่ได้ 123 คือค่าของตัวแปร Session x และ 456 คือค่าตัวแปร Session y ซึ่งตัวแปรทั้งสองได้ประกาศไว้ใน โปรแกรม 1.php ไว้แล้ว

ผลลัพธ์ เมื่อรันโปรแกรม 2.php โดยไม่ผ่าน โปรแกรม 1.php

[Back](#)

อธิบายผลลัพธ์

ผลลัพธ์ จะไม่แสดงค่าของตัวแปร x และ y ออกมา เพราะไม่มีการลงทะเบียนไว้ก่อน

ระบบสมาชิก

การประยุกต์ใช้งานตัวแปร Session จะใช้เพื่อเก็บตัวแปร User ไว้ แล้วถ้าเปิดหน้าเว็บโดยไม่ผ่านหน้า Log In โปรแกรมจะทำการตรวจสอบก่อน แล้วเปิดหน้า Log In ให้ Log In ก่อน และเมื่อทำการ Log In โปรแกรมจะทำการลงทะเบียนตัวแปร Session ไว้ และ ทำลายตัวแปร Session เมื่อทำการออกจากระบบ

ฐานข้อมูลเพื่อเก็บข้อมูลสมาชิก

สร้างฐานข้อมูลเพื่อข้อมูล User โดยเก็บรายละเอียดของผู้ใช้ คือ Username , Password , ชื่อ , เพศ , วันเกิด , ที่อยู่ , งานอดิเรก , E-mail , ประเทศ , คำถามและคำตอบ กรณีลืมรหัสผ่าน ดังนี้

	Field	Type	Collation
☐	mem_id	int(11)	
☐	mem_username	varchar(10)	tis620_thai_ci
☐	mem_password	varchar(10)	tis620_thai_ci
☐	mem_name	varchar(10)	tis620_thai_ci
☐	mem_gender	varchar(10)	tis620_thai_ci
☐	mem_birthday	date	
☐	mem_address	mediumtext	tis620_thai_ci
☐	mem_hobby	varchar(255)	tis620_thai_ci
☐	mem_email	varchar(30)	tis620_thai_ci
☐	mem_country	varchar(50)	tis620_thai_ci
☐	mem_question	varchar(255)	tis620_thai_ci
☐	mem_answer	varchar(255)	tis620_thai_ci

รูปที่ 7-1 : แสดงข้อมูลรายละเอียดสมาชิก
ตัวอย่างที่ 7-3 โปรแกรม Login.html

```

1  <FORM METHOD=POST ACTION="login.php">
2  <TABLE border=1>
3  <TR>
4    <TD>User :: </TD>
5    <TD><INPUT TYPE="text" NAME="txtUser"></TD>
6  </TR>
7  <TR>
8    <TD>Password :: </TD>
9    <TD><INPUT TYPE="password" NAME="txtPass"></TD>
10 </TR>
11 </TABLE>
12 <INPUT TYPE="submit" Value = "Send">
13 <INPUT TYPE="reset" value = "Cancel"> <br>
14 <A HREF="registerform.php">Register</A><BR>
15 <A HREF="forgot.html">Forgot Password</A>
▶16 </FORM>

```

ผลลัพธ์

รูปที่ 7-2 : แสดงหน้าเข้าระบบ

อธิบายรหัสคำสั่ง

บรรทัดที่ 1-12 เป็นคำสั่งสร้างฟอร์ม เพื่อรับข้อมูล User และ Password และมีปุ่ม Submit และ Reset

บรรทัดที่ 14 เป็นคำสั่ง Link ไปที่ RegisterForm.php กรณีที่ต้องการสมัครสมาชิก

บรรทัดที่ 15 เป็นคำสั่ง Link ไปที่ Forgot.html กรณีที่ลืมรหัสผ่าน

ตัวอย่างที่ 7-4 โปรแกรม Connect.php

```

1 <META http-equiv="Content-type content/html;charset=TIS-620">
2 <!-- connect.php -->
3 <?
4 #1. connect HOST
5 $conn = mysql_connect("localhost","root","1234") or die ("Access Host denied.");
6 # 2. connect DATABASE
7 mysql_select_db("MIS",$conn) or die ("Access Database denied.");
8 ?>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นคำสั่ง Meta tag เพื่อแสดงข้อมูลเป็นภาษาไทย

บรรทัดที่ 2 เป็นหมายเหตุของภาษา HTML

บรรทัดที่ 3 เครื่องหมาย <? หมายถึง เปิดคำสั่ง PHP

บรรทัดที่ 4 เป็นหมายเหตุของภาษา PHP

บรรทัดที่ 5 เป็นการติดต่อเครื่องแม่ข่าย จะกำหนดไว้ในไฟล์ connect.php เพราะเครื่องแม่ข่ายจะเป็นเครื่องเดิม และถ้าเปลี่ยนเครื่องแม่ข่าย เช่นขึ้น Host จริง ก็เปลี่ยนข้อมูลเครื่องแม่ข่ายเพียงไฟล์ connect.php ไฟล์เดียว

บรรทัดที่ 6 เป็นหมายเหตุของภาษา PHP

บรรทัดที่ 7 เป็นคำสั่ง เพื่อติดต่อฐานข้อมูล ชื่อ MIS ถ้าติดต่อไม่ได้ จะแสดงข้อความว่า Access Database denied

ตัวอย่างที่ 7-5 โปรแกรม Login.php

```

1  <? ob_start();
2  include ("connect.php");
3  $sql = "select mem_username,mem_password,mem_name from member where
4  mem_username like '$txtUser' and mem_password like '$txtPass' ";
5  $result = mysql_query($sql) or die(mysql_error());
6  $totalrows=mysql_num_rows($result);
7  $rs = mysql_fetch_array($result);
8  if ($totalrows > 0) {
9      echo "Welcome Khun ". $rs['mem_name'] ;
10     session_start();
11     session_register('username');
12     $_SESSION['username'] = $rs['mem_username'];
13     header("Location:index.php");
14 }
15 else {
16     echo "<Script>alert ('Login Fail !!! ');    history.back();    </script> ";
17 }
18 mysql_close($conn);
19 ob_end_flush();
20 ?>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 ob_start() เป็นคำสั่งที่บอกไปยัง webserver ว่า อย่าเพิ่งส่ง output ไปให้ client นะ รวบรวมไว้ก่อน รอให้ส่งหรือประมวลผลไปถึงบรรทัดสุดท้าย ก่อนแล้วค่อยส่ง ไปให้ Client

บรรทัดที่ 2 เป็นการดึง Code จากไฟล์ Connect.php มาประมวลผลด้วย แต่เนื่องด้วยการทำงานในไฟล์ Connect คือ ขั้นตอนการติดต่อ เครื่องแม่ข่ายและฐานข้อมูล ซึ่งเป็นขั้นตอนที่ใช้เหมือนกันทุกครั้งจึงเก็บไว้ในไฟล์ Connect ไฟล์ เดียว และง่ายต่อการแก้ไขถ้าต้องการเปลี่ยนเครื่อง หรือเปลี่ยน Host

บรรทัดที่ 3-4 เป็นคำสั่งเพื่อดึงข้อมูลในฐานข้อมูล ในตาราง member โดยดึง Record ที่มี Username และ Password ตรงกับที่คีย์เข้ามาใน Form Log IN

บรรทัดที่ 5 เป็นคำสั่งรันคำสั่ง SQL ที่เขียนไว้

บรรทัดที่ 6 เป็นคำสั่งดึงจำนวนแถวออกมาว่าได้กี่แถวจากการรันคำสั่ง SQL

บรรทัดที่ 7 เป็นคำสั่งดึงแถวแรกของการรันคำสั่ง SQL

บรรทัดที่ 8 เป็นการตรวจสอบว่าถ้ามีจำนวนแถว มากกว่า 0 แสดงว่า ชื่อและรหัสผ่านถูก

บรรทัดที่ 9 เป็นคำสั่งแสดง ชื่อ ของผู้ใช้

บรรทัดที่ 10 เป็นคำสั่งเริ่มใช้ตัวแปร \$_SESSION

บรรทัดที่ 11 เป็นคำสั่งลงทะเบียน ตัวแปร \$_SESSION username

บรรทัดที่ 12 เป็นคำสั่งกำหนดค่า ตัวแปร \$_SESSION username ให้มีค่าเท่ากับ ค่า User Name

บรรทัดที่ 13 เป็นคำสั่งให้แสดงการทำงานไปที่ไฟล์ index.php

บรรทัดที่ 15 เป็นคำสั่ง Java Script ให้ข้อความ Login Fail และกลับไปไฟล์เดิม คือไฟล์ Login.html

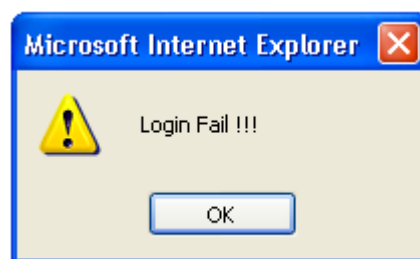
บรรทัดที่ 18 เป็นคำสั่ง ปิดการเชื่อมต่อเครื่องแม่ข่าย

บรรทัดที่ 19 เป็นคำสั่ง ให้ส่งข้อมูลในบัพเฟอร์ แล้วปิดบัพเฟอร์

หมายเหตุ

บรรทัดที่ 13 สามารถแทนด้วยคำสั่ง `<meta http-equiv="refresh" content="0;url=index.php">` ได้

ผลลัพธ์ กรณีที่ ชื่อ ผู้ใช้ หรือ รหัสผ่านไม่ถูกต้อง



รูปที่ 7-3 : แสดงผลลัพธ์เข้าระบบไม่สำเร็จ

ผลลัพธ์ ที่สามารถเข้าระบบได้ จะแสดงผลไฟล์ Index.php



รูปที่ 7-4 : แสดงผลลัพธ์เข้าระบบได้

ตัวอย่างที่ 7-6 โปรแกรม Index.php

```

1  <?
2  session_start();
3  if (isset($_SESSION['username'])) header("Location:login.html");
4  ?>
5  Welcome member<br>
6  <TABLE BORDER=1>
7  <TR><TD>แก้ไขข้อมูล </TD><TD><A HREF="edit.php">Edit Profile</A></TD> </TR>
8  <TR><TD>เปลี่ยนรหัสลับ</TD><TD><A HREF="change.php">Change Password</A></TD></TR>
9  <TR><TD>ออกจากระบบ</TD><TD><A HREF="logout.php">Logout</A></TD></TR>
10 </TABLE>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เครื่องหมาย <? หมายถึง เปิดคำสั่ง PHP

บรรทัดที่ 2 เป็นคำสั่งเริ่มใช้ตัวแปร \$_SESSION

บรรทัดที่ 3 เป็นคำสั่ง ที่ตรวจสอบว่า มีการลงทะเบียนตัวแปร \$_SESSION username แล้ว หรือยังถ้ายังแสดงว่ายังมาได้ Login ให้ไปเปิดหน้า Login.html

บรรทัดที่ 4 เครื่องหมาย ?> หมายถึง ปิดคำสั่ง PHP

บรรทัดที่ 5 แสดงข้อความ Welcome member

บรรทัดที่ 6 เปิดคำสั่ง แสดงตาราง โดยมีเส้นขอบเท่ากับ 1

บรรทัดที่ 7 เป็นคำสั่งแสดงข้อความ แก้ไขข้อมูล และเมื่อคลิกจะ Link ไปที่ไฟล์ Edit.php

บรรทัดที่ 8 เป็นคำสั่งแสดงข้อความ เปลี่ยนรหัสผ่าน เมื่อคลิกจะ Link ไปที่ไฟล์ Change.php

บรรทัดที่ 9 เป็นคำสั่งแสดงข้อความ ออกจากโปรแกรม เมื่อคลิกจะ Link ไปที่ไฟล์ Logout.php

ตัวอย่างที่ 7-7 โปรแกรม Logout.php

```

1  <?
2  session_start();
3  session_destroy();
4  ?>
5  Bye bye
6  <A HREF="login.html">Login Again</A>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เครื่องหมาย <? หมายถึง เปิดคำสั่ง PHP

บรรทัดที่ 2 เปิดคำสั่งเริ่มการทำงานตัวแปร Session

บรรทัดที่ 3 เปิดคำสั่งยกเลิกตัวแปร Session

บรรทัดที่ 4 เครื่องหมาย ?> หมายถึง ปิดคำสั่ง PHP

บรรทัดที่ 5 เป็นการแสดงข้อความ Bye Bye

บรรทัดที่ 7 เป็นการแสดงข้อความ Login Again แล้วลิงค์ไปที่ Login.html

ตัวอย่างที่ 7-8 โปรแกรม RegisterForm.php

```

1 <SCRIPT LANGUAGE="JavaScript">
2   function check(frm) {
3     if (frm.txtUser.value.length < 1) {
4       alert ("Please Input UserName");
5       frm.txtUser.focus();
6       return false;   }
7     if (frm.txtPass.value.length < 1) {
8       alert ("Please Input Password");
9       frm.txtPass.focus();
10      return false;   }
11     if (frm.txtConf.value == "") {
12       alert ("Please Input Confirm Password");
13       frm.txtConf.focus();
14       return false;   }
15     if (frm.txtConf.value != frm.txtPass.value) {
16       alert ("Password Mismatch ");
17       frm.txtPass.value = "";
18       frm.txtConf.value = "";
19       frm.txtPass.focus();
20       return false;   }
21     if (frm.txtName.value == "") {
22       alert ("Please Input Name");
23       frm.txtName.focus();
24       return false;   } } </SCRIPT>
25 <body onload="document.x.txtUser.focus()">
26 <FORM METHOD=POST ACTION="register.php" onSubmit="return check(this)" name="x">
27 <TABLE border=1>
28 <TR>
29   <TD>User ::</TD>
30   <TD><INPUT TYPE="text" NAME="txtUser"></TD>
31 </TR>
32 <TR>
33   <TD>Password ::</TD>
34   <TD><INPUT TYPE="password" Name = "txtPass"></TD>
35 </TR>
36 <TR>
37   <TD>Confirm Password ::</TD>
38   <TD><INPUT TYPE="password" NAme= "txtConf"></TD>
39 </TR>
40 <TR>
41   <TD>Name ::</TD>
42   <TD><INPUT TYPE="text" NAME="txtName"></TD>
43 </TR>
44 <TR>
45   <TD>Gender ::</TD>
46   <TD><INPUT TYPE="radio" NAME="Sex" value = "F">Female
47   <INPUT TYPE="radio" NAME="Sex" value = "M">Male
48   </TD>
49 </TR>
50 <TR>
51   <TD>Address ::</TD>
52   <TD><TEXTAREA NAME="address" ROWS="2" COLS="20"></TEXTAREA> </TD>
53 </TR>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นคำสั่ง เริ่มต้นเปิด Java Script

บรรทัดที่ 2 เป็นคำสั่ง สร้างฟังก์ชัน check

บรรทัดที่ 3 เป็นคำสั่ง ตรวจสอบว่า ช่อง User มีการคีย์ข้อมูลหรือไม่ โดยตรวจสอบความยาวของ Text Box

บรรทัดที่ 4 เป็นคำสั่ง เพื่อให้กล่องข้อความเตือนว่า Please Input User name

บรรทัดที่ 5 เป็นคำสั่ง เพื่อให้ Curser รองรับข้อมูลที่ txtUser

บรรทัดที่ 6 เป็นคำสั่ง เพื่อส่งผลลัพธ์ค่า False ให้กับฟังก์ชัน check

บรรทัดที่ 11 เป็นคำสั่ง ตรวจสอบว่า ช่องยืนยันรหัสผ่านมีการคีย์ข้อมูลหรือไม่ โดยตรวจสอบค่าว่าง

บรรทัดที่ 15-19 เป็นคำสั่ง ตรวจสอบว่าช่องรหัสผ่านกับช่องยืนยันรหัสผ่านมีการคีย์ข้อมูลตรงกันหรือไม่ ถ้าไม่ตรงให้ลบข้อความทั้งสองช่องออก แล้วรอรับการคีย์ข้อมูลใหม่ และส่งผลลัพธ์เป็น False

บรรทัดที่ 25 เป็นคำสั่ง เพื่อให้เปิดเอกสาร x มาครั้งแรกให้แสดง Curser ที่ txtUser เลย

บรรทัดที่ 25 onSubmit เป็นการสั่งให้โปรแกรมไปทำงานที่ ฟังก์ชัน Check ก่อน มีมีการกดปุ่ม Submit และ name เป็นการตั้งชื่อเอกสาร เพื่อใช้ร่วมกับบรรทัดที่ 25 เพื่อใช้กับ onload

บรรทัดที่ 57 เป็นคำสั่งเพื่อให้ ค่าตัวแปร \$country เก็บรายชื่อประเทศไว้

บรรทัดที่ 58-61 เป็นคำสั่งเพื่อให้ List Box แสดงค่าในตัวแปร \$country ออกมา

บรรทัดที่ 70-84 เป็นคำสั่งเพื่อให้ List Box แสดงค่าเป็นวัน เดือน และ ปี

```

53 </TR>
54 <TR>
55   <TD>Country :: </TD>
56   <TD>
57     <? $country = Array("America","Australia","Thailand") ?>
58     <SELECT NAME="txtCountry">
59       <? for($i=0;$i<sizeof($country); $i++){ ?>
60         <Option Value = "<?=$country[$i] ?>"><?=$country[$i]?>
61         <?}?>
62     </SELECT> </TD>
63 </TR>
64 <TR>
65   <TD>Email :: </TD>
66   <TD> <INPUT TYPE="text" NAME="txtEmail"></TD>
67 </TR>
68 <TR>
69   <TD>Birthday :: </TD>
70   <TD><SELECT NAME="txtDay">
71     <? for($i=1;$i<=31; $i++){ ?>
72       <Option Value = "<?=$i?>"><?=$i?>
73       <?}?>
74     </SELECT>
75     <SELECT NAME="txtMonth">
76       <? $mnt = Array("Jan","Feb","Mar","Apr","May","Jun") ?>
77       <? for($i=0;$i<sizeof($mnt); $i++){ ?>
78         <Option Value = "<?=$i + 1 ?>"><?=$mnt[$i]?>
79         <?}?>
80     </SELECT>
81     <SELECT NAME="txtYear">
82       <? for($i=Date("\Y");$i>=Date("\Y")-100; $i--){ ?>
83         <Option Value = "<?=$i?>"><?=$i?>
84         <?}?>
85     </SELECT>
86   </TD>
87 </TR>
88 <TR>
89   <TD>Hobby :: </TD>
90   <TD>
91     <INPUT TYPE="checkbox" NAME="hobby[]" value = "Reading">Reading<BR>
92     <INPUT TYPE="checkbox" NAME="hobby[]" value = "Swimming">Swimming<BR>
93     <INPUT TYPE="checkbox" NAME="hobby[]" value = "Game">Game<BR>
94   </TD>
95 </TR>
96 <TR>
97   <TD>Question :: </TD>
98   <TD> <INPUT TYPE="text" NAME="txtQuestion">
99   </TD>
100 </TR>
101 <TR>
102   <TD>Answer :: </TD>
103   <TD> <INPUT TYPE="text" NAME="txtAnswer">
104   </TD>
105 </TR>
106 </TABLE>
107 <INPUT TYPE="submit" Value = "Send">
108 <INPUT TYPE="reset" Value = "Cancel">
109 </FORM>

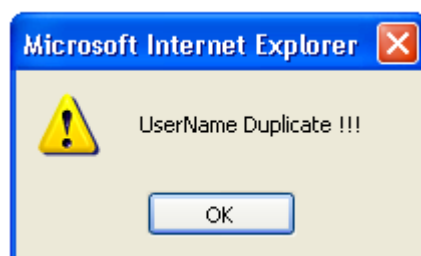
```

ผลลัพธ์ กรณีที่ไม่กรอกข้อมูลช่อง **UserName**



รูปที่ 7-5 : แสดงผลลัพธ์การรหัสผู้ใช้ไม่ถูกต้อง

ผลลัพธ์ กรณีที่ชื่อ **UserName** ซ้ำในฐานข้อมูล



รูปที่ 7-6 : แสดงผลลัพธ์การรหัสผู้ใช้ซ้ำ

ตัวอย่างที่ 7-9 โปรแกรม Register.php

```

1  <?
2  include ("connect.php");
3  $sql = "select mem_username from member where mem_username like '$txtUser' ";
4  $result = mysql_query($sql) or die(mysql_error());
5  $totalrows=mysql_num_rows($result);
6  if ($totalrows > 0) {
7      echo "<Script>alert ('UserName Duplicate !!! ');   history.back();    </script> ";
8      exit;
9  }
10 $temp = $_POST['hobby'];
11 $hobby="";
12 for ($i=0; $i<sizeof($temp); $i++){
13     $hobby.="$temp[$i].";
14 }
15 $hobby = substr($hobby,0,-1);
16 $sql = "insert into member values
17 ('.$_POST['txtUser']).'$_POST['txtPass']).'$_POST['txtName']).'$_POST['Sex']).'
18 '$_POST['txtYear']).'$_POST['txtMonth']).'$_POST['txtDay']).'$_POST['address']).'
19 '$_POST['txtEmail']).'$_POST['txtCountry']).'
20 '$_POST['txtQuestion']).'$_POST['txtAnswer']).'";
21 echo $sql."<br>";
22 $result = mysql_query($sql) or die(mysql_error());
23 if ($result) echo "INSERT Complete"; else echo "Cannot Insert data";
24 }
▶25 mysql_close($conn);

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นคำสั่ง เริ่มต้นเปิดเอกสาร PHP

บรรทัดที่ 2 เป็นคำสั่ง ดึงรหัสคำสั่งมาจากไฟล์ connect.php

บรรทัดที่ 3 การกำหนดคำสั่ง SQL เพื่อตรวจสอบว่าในฐานข้อมูลมี User ชื่อเดียวกันหรือไม่

บรรทัดที่ 4 รันคำสั่งคำสั่ง SQL เพื่อตรวจสอบว่าในฐานข้อมูลมี User ชื่อเดียวกันหรือไม่

บรรทัดที่ 5 การดึงจำนวนของผลลัพธ์ จากการตรวจสอบว่าในฐานข้อมูลมี User นี้กี่คน

บรรทัดที่ 6-9 คำสั่งเพื่อตรวจสอบจำนวนว่ามากกว่า 0 หรือไม่ ถ้ามากกว่าให้แสดง Message Box ว่า UserName Duplicate แล้วกลับไปรับข้อมูลใหม่

บรรทัดที่ 10-14 คำสั่งเก็บค่างานอดิเรกที่เช็คจากหน้ากรอกข้อมูล แล้วเก็บเป็นข้อความค้น ด้วยเครื่องหมาย ,

บรรทัดที่ 15 คำสั่งที่ตัด String จากข้างหลัง มาหนึ่งตัว (-1) เพราะเครื่องหมาย , เกินมา 1 ตัว

บรรทัดที่ 16-20 คำสั่งเพิ่มข้อมูลจากที่กรอกเข้ามาผ่าน Method Post เก็บลงฐานข้อมูลตาราง member

บรรทัดที่ 22 คำสั่งรันรหัสคำสั่ง SQL แล้วเก็บผลลัพธ์ไว้ที่ตัวแปร Result

บรรทัดที่ 23 คำสั่งเพื่อตรวจสอบว่า Insert ข้อมูลได้หรือไม่ ถ้าได้แสดงว่า Insert Complete ถ้าไม่ได้ แสดงว่า Cannot Insert Data

บรรทัดที่ 25 คำสั่งเพื่อยกเลิกการติดต่อเครื่องแม่ข่าย

ตัวอย่างที่ 7-10 โปรแกรม Forgot.html

```

1 <FORM METHOD=POST ACTION="showq.php">
2 <TABLE border=1>
3 <TR>
4   <TD>User :: </TD>
5   <TD><INPUT TYPE="text" NAME="txtUser"></TD>
6 </TR>
7 </TABLE>
8 <INPUT TYPE="submit" Value = "Send">
9 <INPUT TYPE="reset" value ="Cancel"> <br>
10 </FORM>

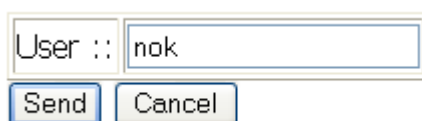
```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นคำสั่ง เริ่มต้นการสร้าง FORM โดยกดปุ่ม Submit แล้วจะไปที่ไฟล์ showq.php

บรรทัดที่ 2-9 เป็นคำสั่งเพื่อรับ User ที่ต้องการค้นหารหัสผ่าน

ผลลัพธ์



The screenshot shows a web form with a label 'User ::' followed by a text input field containing the text 'nok'. Below the input field are two buttons: 'Send' and 'Cancel'.

รูปที่ 7-7 : แสดงหน้าจอกรณีลืมรหัสผ่าน

ตัวอย่างที่ 7-11 โปรแกรม ShowQ.php

```

1  <?
2  include ("connect.php");
3  $sql = "select * from member where mem_username like '$txtUser' ";
4  $result = mysql_query($sql) or die(mysql_error());
5  $totalrows=mysql_num_rows($result);
6  if ($totalrows > 0) {
7  $rs = mysql_fetch_array($result);
8  ?>
9  <FORM METHOD=POST ACTION="showa.php?id=<?=$rs[0]?>">
10 <TABLE border=1>
11 <TR>
12   <TD>Question :: </TD>
13   <TD><?=$rs['mem_question'] ?></TD>
14 </TR>
15 <TR>
16   <TD>Answer :: </TD>
17   <TD><INPUT TYPE="text" NAME="txtAns"></TD>
18 </TR>
19 </TABLE>
20 <INPUT TYPE="submit" Value = "Send">
21 <INPUT TYPE="reset" value = "Cancel"> <br>
22 </FORM>
23 <?
24 }
25 else {
26   echo "<Script>alert ('User Incorrect !!! '); history.back();   </script> ";
27 }
28 mysql_close($conn);
29 ?>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 3 กำหนดคำสั่ง SQL เพื่อดึงระเบียบในฐานข้อมูลมี User ตรงกับที่กรอกมาหรือไม่

บรรทัดที่ 4 การรันคำสั่งคำสั่ง SQL

บรรทัดที่ 5 การดึงจำนวนของผลลัพธ์ จากการตรวจสอบว่าในฐานข้อมูลมี User นี้กี่คน

บรรทัดที่ 6-8 คำสั่งเพื่อตรวจสอบว่ามี User นี้ในฐานข้อมูลหรือไม่ ถ้ามีให้ดึงข้อมูลมาใส่ไว้ใน \$rs

บรรทัดที่ 9 คำสั่งเพื่อสร้างฟอร์ม โดยกำหนดว่าเมื่อกดปุ่ม Submit แล้วจะไปทำงานที่ไฟล์ showa.php พร้อมกับส่งรหัส Id ไปพร้อมกับ url ด้วย

บรรทัดที่ 13 คำสั่งเพื่อแสดงคำถามของ user นี้ มาแสดง

บรรทัดที่ 26 คำสั่งกรณี que user name ไม่ถูกต้อง คือไม่มีในฐานข้อมูล

ผลลัพธ์

Question ::	Are you full?
Answer ::	
Send	Cancel

ตัวอย่างที่ 7-12 โปรแกรม ShowA.php

```

1  <?
2  include ("connect.php");
3  $id = $_GET['id'];
4  $sql = "select * from member where
5  mem_answer like '{$_POST['txtAns']}' and mem_id = '$id' ";
▶ 6  $result = mysql_query($sql) or die(mysql_error());
7  $totalrows=mysql_num_rows($result);
8  if ($totalrows > 0) {
9      $rs = mysql_fetch_array($result);
10     echo "Your Password is ". $rs['mem_password'] ;
11 }
12 else {
13     echo "<Script>alert ('Answer Incorrect !!! '); history.back(); </script> ";
14 }
15 mysql_close($conn);
16 ?>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นคำสั่ง เริ่มต้นเปิดเอกสาร PHP

บรรทัดที่ 2 คำสั่งดึงรหัสคำสั่งมาจากไฟล์ connect.php

บรรทัดที่ 3 คำสั่งดึงค่าตัวแปร id ด้วย method GET

บรรทัดที่ 4-5 กำหนดคำสั่ง SQL เพื่อตรวจสอบว่าคำตอบที่ได้รับมากับฐานข้อมูลถูกต้องหรือไม่

บรรทัดที่ 6 การรันคำสั่งคำสั่ง SQL

บรรทัดที่ 7 การดึงจำนวนของผลลัพธ์ จากการตรวจสอบว่าในฐานข้อมูลว่ามีเท่าไร

บรรทัดที่ 9-10 คำสั่งเพื่อตรวจสอบว่าคำตอบถูกต้องหรือไม่ ถ้าถูกให้ดึงข้อมูลมาใส่ไว้ใน \$rs และแสดง Password ออกมาแสดง

บรรทัดที่ 13 คำสั่งแสดง กรณีที่คำตอบไม่ถูกต้อง จะแสดง ข้อความว่า Answer Incorrect

บรรทัดที่ 15 คำสั่งเพื่อยกเลิกการติดต่อเครื่องแม่ข่าย

ผลลัพธ์ กรณีที่ คำตอบถูกต้อง

Your Password is 999

ผลลัพธ์ กรณีที่ คำตอบถูกต้อง

Your Password is 999

ตัวอย่างที่ 7-13 โปรแกรม Edit.php

```

1  <?
2  session_start();
3  if (isset($_SESSION['username'])) header("Location:login.html");
4  ?>
5  <? ob_start();
6  include ("connect.php");
7  error_reporting(E_ALL ^ E_NOTICE);
8  $sql = "select *from member where mem_username like '{$_SESSION['username']}' ";
9  $result = mysql_query($sql) or die(mysql_error());
10 $totalrows=mysql_num_rows($result);
11 $rs = mysql_fetch_array($result);
12 ?>
13 <SCRIPT LANGUAGE="JavaScript">
14     function check(frm) {
15         if (frm.txtName.value == "") {
16             alert ("Please Input Name");
17             frm.txtName.focus();
18             return false;
19         }
20     }
21 </SCRIPT>
22 <body onload="document.x.txtUser.focus()">
23 <FORM METHOD=POST ACTION="update.php" onSubmit="return check(this)" name="x">
24 <TABLE border=1>
25 <TR>
26     <TD>User ::</TD>
27     <TD><INPUT TYPE="text" NAME="txtUser" value = "<?=$rs['mem_username']?>" readonly></TD>
28 </TR>
29 <TR>
30     <TD>Name ::</TD>
31     <TD><INPUT TYPE="text" NAME="txtName" value = "<?=$rs['mem_name']?>"></TD>
32 </TR>
33 <TR>

```

```

34 <TD>Gender :: </TD> <TD>
35 <INPUT TYPE="radio" NAME="Sex" value = "F" <?if ($rs['mem_gender'] == "F") echo "checked"?>>Female
36 <INPUT TYPE="radio" NAME="Sex" value = "M" <?if ($rs['mem_gender'] == "M") echo "checked"?>>Male
37 </TD>
38 </TR>
39 <TR>
40 <TD>Address :: </TD>
41 <TD><TEXTAREA NAME="address" ROWS="2" COLS="20"><?=$rs['mem_address']?></TEXTAREA>
42 </TD></TR>
43 <TR>
44 <TD>Country :: </TD>
45 <TD>
46 <? $country = Array("America","Australia","Thailand") ?>
47 <SELECT NAME="txtCountry">
48 <? for($i=0;$i<sizeOf($country); $i++){ ?>
49 <Option Value = "<?=$country[$i] ?>"
50 <? if ($rs['mem_country'] == $country[$i]) echo "Selected"?>
51 ><?=$country[$i]?>
52 <?}?>
53 </SELECT> </TD>
54 </TR>
55 <TR>
56 <TD>Email :: </TD>
57 <TD><INPUT TYPE="text" NAME="txtEmail" value="<?=$rs['mem_email']?>"></TD>
58 </TR>
59 <TR>
60 <TD>Birthday :: </TD>

```

```

61
62 <TD><? $t = strtotime($rs['mem_birthday']);
63 $d = Date("d",$t);
64 $m = Date("m",$t);
65 $y = Date("Y",$t); ?>
66 <SELECT NAME="txtDay">
67 <? for($i=1;$i<=31; $i++){ ?>
68 <Option Value = "<?=$i?>" <? if ($d==$i) echo "Selected"?>><?=$i?>
69 <?}?>
70 </SELECT>
71 <SELECT NAME="txtMonth">
72 <? $mnt = Array("Jan","Feb","Mar","Apr","May","Jun","Jul","Aug","Sep","Oct","Nov","Dec") ?>
73 <? for($i=0;$i<sizeOf($mnt); $i++){ ?>
74 <Option Value = "<?=$i + 1 ?>" <? if ($m==$i+1) echo "Selected"?>><?=$mnt[$i]?>
75 <?}?>
76 </SELECT>
77 <SELECT NAME="txtYear">
78 <? for($i=Date("Y");$i>=Date("Y")-100; $i--){ ?>
79 <Option Value = "<?=$i?>" <? if ($y==$i) echo "Selected"?>><?=$i?>
80 <?}?>
81 </SELECT>
82 </TD>
83 </TR>
84 <TR>
85 <TD>Hobby :: </TD>
86 <TD>

```

```

87     <INPUT TYPE="checkbox" NAME="hobby[]" value = "Reading"
88     <? if (strstr($rs['mem_hobby'], "Reading")) echo "checked"?>>Reading<BR>
89     <INPUT TYPE="checkbox" NAME="hobby[]" value = "Swimming"
90     <? if (strstr($rs['mem_hobby'], "Swimming")) echo "checked"?>>Swimming<BR>
91     <INPUT TYPE="checkbox" NAME="hobby[]" value = "Game"
92     <? if (strstr($rs['mem_hobby'], "Game")) echo "checked"?>>Game<BR>
93     </TD>
94 </TR>
95 <TR>
96     <TD>Question :: </TD>
97     <TD> <INPUT TYPE="text" NAME="txtQuestion" value = "<?=$rs['mem_question']?>">
98     </TD>
99 </TR>
100 <TR>
101     <TD>Answer :: </TD>
102     <TD> <INPUT TYPE="text" NAME="txtAnswer" value = "<?=$rs['mem_answer']?>">
103     </TD>
104 </TR>
105 </TABLE>
106 <INPUT TYPE="submit" Value = "Send">
107 <INPUT TYPE="reset" Value = "Cancel">
108 </FORM>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1-4 เป็นคำสั่งเพื่อตรวจสอบการ Login ว่ามีการ Login หรือยังถ้ายังให้ไป Login ก่อน

บรรทัดที่ 7 คำสั่งเพื่อกำหนดการแสดงผลของข้อมูลให้แสดงข้อมูลผิดพลาด ยกเว้น ค่าเตือนไม่ต้องแสดง

บรรทัดที่ 8-12 คำสั่งดึงข้อมูลของผู้ใช้ มาไว้ในตัวแปร \$rs

บรรทัดที่ 13-21 คำสั่ง Script เพื่อตรวจสอบการคืนข้อมูล ชื่อ

บรรทัดที่ 22 เป็นคำสั่ง เพื่อให้เปิดเอกสาร x มาครั้งแรกให้แสดง Cursor ที่ txtUser เลย

บรรทัดที่ 23 onSubmit เป็นการสั่งให้โปรแกรมไปทำงานที่ ฟังก์ชัน Check ก่อน มีมีการกดปุ่ม Submit และ name เป็นการตั้งชื่อเอกสาร เพื่อใช้ร่วมกับบรรทัดที่ 25 เพื่อใช้กับ onload

บรรทัดที่ 27 การแสดงผลข้อมูล User name โดยกำหนดเป็น ReadOnly คือ ไม่สามารถแก้ไขได้

บรรทัดที่ 35-36 วิธีการแสดงผลข้อมูลเพช หรือ Radio Button โดยกำหนดให้ เช็ค Box หรือไม่ตามค่าที่ปรากฏในตัวแปร โดยใช้คำสั่ง Checked

บรรทัดที่ 46-52 วิธีการแสดงผลข้อมูลประเทศ แบบ List Box โดยถ้าข้อมูลเป็นประเทศใดจะแสดงประเทศนั้น โดยใช้คำสั่ง if และ selected

บรรทัดที่ 62-80 วิธีการแสดงผลข้อมูลแบบวันที่ แบบ List Box โดยถ้าข้อมูลเป็นวันใดจะแสดงวันนั้นออกมา โดยใช้คำสั่ง if และ selected

บรรทัดที่ 87-92 วิธีการแสดงข้อมูลงานอดิเรกแบบ Check Box โดยถ้าข้อมูลเป็นอะไรจะแสดงออกมา โดยใช้คำสั่ง if และ checked

ตัวอย่างที่ 7-14 โปรแกรม Update.php

```

1  <?
2  include ("connect.php");
3  error_reporting(E_ALL ^ E_NOTICE);
4  $temp = $_POST['hobby'];
5  for ($i=0; $i<sizeof($temp); $i++){
6      $hobby.=" $temp[$i].";
7  }
8  $hobby = substr($hobby,0,-1);
9  $sql = "update member
10      set mem_name = '{$_POST['txtName']}',
11          mem_gender = '{$_POST['Sex']}',
12          mem_birthday = '{$_POST['txtYear']}-{$_POST['txtMonth']}-{$_POST['txtDay']}',
13          mem_address = '{$_POST['address']}',
14          mem_email = '{$_POST['txtEmail']}',
15          mem_hobby = '$hobby',
16          mem_country = '{$_POST['txtCountry']}',
17          mem_question = '{$_POST['txtQuestion']}',
18          mem_answer = '{$_POST['txtAnswer']}'
19      where mem_username = '{$_POST['txtUser']}' ";
20  echo $sql."<br>";
21  echo "<br>";
22  $result = mysql_query($sql) or die(mysql_error());
23  if ($result) echo "Update Completed"; else echo "Cannot Update data";
24  mysql_close($conn)
25  ?>
26  <br>
27  <A HREF="login.html">HOME</A>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นคำสั่ง เริ่มต้นเปิดเอกสาร PHP

บรรทัดที่ 2 คำสั่งเพื่อกำหนดการแสดงผลข้อมูลให้แสดงข้อมูลผิดพลาด ยกเว้น ค่าเตือนไม่ต้องแสดง**บรรทัดที่ 4-7** คำสั่งเก็บค่างานอดิเรกที่เช็คจากหน้ากรอกข้อมูล แล้วเก็บเป็นข้อความค้นด้วยเครื่องหมาย ,

บรรทัดที่ 8 คำสั่งที่ตัด String จากข้างหลัง มาหนึ่งตัว (-1) เพราะเครื่องหมาย , เกินมา 1 ตัว

บรรทัดที่ 9-19 กำหนดคำสั่ง SQL เพื่อ Update ข้อมูลตามที่ได้กรอกมา ในรหัสผู้ใช้ตรงกัน

บรรทัดที่ 22 การรันคำสั่งคำสั่ง SQL

บรรทัดที่ 23 คำสั่งเพื่อตรวจสอบว่าแก้ไขข้อมูลได้หรือไม่ ถ้าได้ให้แสดงข้อความว่า Update Complete ถ้าไม่ได้ให้แสดงว่า Cannot Update data

บรรทัดที่ 24 คำสั่งเพื่อยกเลิกการติดต่อเครื่องแม่ข่าย

บรรทัดที่ 27 คำสั่ง Link ไปที่ไฟล์ Login.html

ตัวอย่างที่ 7-15 โปรแกรม Change.php

```

1  <SCRIPT LANGUAGE="JavaScript">
2      function check(frm) {
3          if (frm.txtCurr.value == "") {
4              alert ("Please Input Current Password");
5              frm.txtCurr.focus();
6              return false;
7          }
8          if (frm.txtPass.value == "") {
9              alert ("Please Input new Password");
10             frm.txtPass.focus();
11             return false;
12         }
13         if (frm.txtConf.value == "") {
14             alert ("Please Input Confirm Password");
15             frm.txtConf.focus();
16             return false;
17         }
18         if (frm.txtConf.value != frm.txtPass.value) {
19             alert ("Password Not like Confirm Password");
20             frm.txtConf.focus();
21             return false;
22         }
23     }
24 </SCRIPT>
25 <FORM METHOD=POST ACTION="changepassword.php" onSubmit="return check(this)">
26 Current Password :: <INPUT TYPE="password" name = "txtCurr"><br>
27 New Password :: <INPUT TYPE="password" name = "txtPass"><br>
28 Confirm Password :: <INPUT TYPE="password" name = "txtConf"><br>
29 <INPUT TYPE="submit" value = "Submit">
30 <INPUT TYPE="reset" value = "Reset">
31 </FORM>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1-24 เป็นคำสั่งเพื่อตรวจสอบการคืนข้อมูลว่ามีการคืนข้อมูลหรือไม่

บรรทัดที่ 25 onSubmit เป็นการสั่งให้โปรแกรมไปทำงานที่ ฟังก์ชัน Check ก่อน มีมีการกดปุ่ม Submit

บรรทัดที่ 26-28 คำสั่งเพื่อรับข้อมูล

ผลลัพธ์

Current Password ::

New Password ::

Confirm Password ::

ตัวอย่างที่ 7-16 โปรแกรม ChangePassword.php

```

1  <?
2  session_start();
3  if (!isset($_SESSION['username'])) header("Location:login.html");
4  ?>
5  <? ob_start();
6  include ("connect.php");
7  error_reporting(E_ALL ^ E_NOTICE);
8  $sql = "select * from member where mem_username like '{$_SESSION['username']}' and mem_password like '{$_POST['txtCurr']}'";
9  echo $sql."<br>";
10 $result = mysql_query($sql) or die(mysql_error());
11 $totalrows=mysql_num_rows($result);
12 echo $totalrows."<br>";
13 if ($totalrows >0) {
14     $sql = "update member set mem_password='{$_POST['txtPass']}' where mem_username like '{$_SESSION['username']}' ";
15     $result2 = mysql_query($sql) or die(mysql_error());
16     if ($result2)
17         echo "Password changed";
18     else
19         echo "Password cannot update";
20 }
21 else
22     echo "Current Password not correct";
23 ?>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1-4 เป็นคำสั่งเพื่อตรวจสอบการ Login ว่ามีการ Login หรือยังถ้ายังให้ไป Login ก่อน

บรรทัดที่ 7 คำสั่งเพื่อกำหนดการแสดงผลของข้อมูลให้แสดงข้อมูลผิดพลาด ยกเว้น ค่าเตือนไม่ต้องแสดง

บรรทัดที่ 8 คำสั่งดึงข้อมูลของผู้ใช้ โดยตรวจสอบว่า รหัสผู้ใช้ กับรหัสผ่าน ถูกต้องหรือไม่

บรรทัดที่ 10 การรันคำสั่งคำสั่ง SQL

บรรทัดที่ 11 การดึงจำนวนของผลลัพธ์ จากการตรวจสอบว่าในฐานข้อมูลว่ามีเท่าไร

บรรทัดที่ 13-14 คำสั่งเพื่อตรวจสอบว่าชื่อผู้ใช้และรหัสผ่านถูกต้อง หรือไม่ ถ้าถูกต้อง ให้ Update Password ใหม่แทน Password เก่า

บรรทัดที่ 16 การรันคำสั่งคำสั่ง SQL

บรรทัดที่ 17 คำสั่งตรวจสอบการ Update ข้อมูลในฐานข้อมูล ถ้า Update ได้ถูกต้อง ขึ้นข้อความว่า Password changed ถ้า Update ไม่สมบูรณ์ ขึ้นข้อความว่า Password cannot Update

บรรทัดที่ 16 การแสดงข้อความว่า Current Password not Correct

ผลลัพธ์ กรณีที่ใส่ รหัสผ่านเก่าไม่ถูกต้อง

```
select * from member where mem_username like '1' and mem_password like '9'
0
Current Password not correct
```

ผลลัพธ์ กรณีที่ใส่ รหัสผ่านเก่าถูกต้อง และกรอกรหัสใหม่ ตรงกันทั้งสองช่อง

```
select * from member where mem_username like '1' and mem_password like '1'
1
Password changed
```

พระราชดำรัสเกี่ยวกับการศึกษา ของพระบาทสมเด็จพระเจ้าอยู่หัว

"การศึกษาเป็นเครื่องมืออันสำคัญในการพัฒนาความรู้ ความคิด ความประพฤติ ทัศนคติ ค่านิยม และคุณธรรมของบุคคล เพื่อให้เป็นพลเมืองดีมีคุณภาพ และประสิทธิภาพ การพัฒนาประเทศที่ย่อมทำได้สะดวกราบรื่น ได้ผลที่แน่นอนและรวดเร็ว"

(๒๒ กรกฎาคม ๒๕๒๐)

ในการพัฒนาประเทศนั้น จำเป็นต้องทำตามลำดับขั้นเริ่มด้วยการสร้างพื้นฐาน คือความมีกินมีใช้ของประชาชนก่อน ด้วยวิธีการที่ประหยัด ระมัดระวัง แต่ถูกต้องตามหลักวิชาการ เมื่อพื้นฐานเกิดมั่นคงพอควรแล้ว จึงค่อยสร้างเสริมความเจริญขั้นที่สูงขึ้นตามลำดับต่อไป

พระบรมราโชวาท เมื่อวันที่ 19 กรกฎาคม 2517

พระบาทสมเด็จพระเจ้าอยู่หัว ทรงเน้นย้ำแนวทางการพัฒนาที่อยู่บนพื้นฐานของการพึ่งตนเอง ความพอมีพอกิน พอมีพอใช้และการรู้จักความพอประมาณ ทรงเตือนสติประชาชนไม่ให้ประมาท มีคุณธรรมในการดำรงชีวิต ซึ่งแนวคิดดังกล่าวเป็นที่รู้จักกันดีภายใต้ชื่อว่า เศรษฐกิจพอเพียงทรงเน้นหลักการพึ่งตนเองในระดับครอบครัว ผสานกับความสามัคคี ร่วมแรงร่วมใจกันสร้างความเข้มแข็งในระดับท้องถิ่น ก่อนที่จะเชื่อมโยงสู่ระบบธุรกิจในสังคมภายนอก อีกทั้งยังพระราชทานแนวพระราชดำริ ให้หน่วยงานต่างๆ ทั้งจากภาครัฐและภาคประชาชนเข้าร่วมมือกัน เพื่อศึกษาและพัฒนากิจการเกษตรกรรมของไทยอย่างจริงจัง ด้วยทรงตระหนักดีว่า การเกษตรนี้เองที่เป็นรากฐานอันแท้จริง ให้กับพัฒนาการที่ยั่งยืนของประเทศดังพระราชดำรัสที่ว่า " ถ้าชนบทอยู่ได้ บ้านเมืองก็อยู่ได้" ผลลัพธ์จากสายพระเนตรอันยาวไกลของพระองค์ สำแดงให้เป็นที่ประจักษ์ชัดทุกครั้งที เศรษฐกิจของประเทศตกอยู่ในภาวะวิกฤติ และประชาชนชาวไทยทุกฝ่ายทุกระดับ ต่างก็ต้องหันมาทบทวนทิศทางของการพัฒนาประเทศเท่าที่ผ่านมา

ปรัชญาเศรษฐกิจพอเพียงที่ทรงพระราชทาน จึงนับเป็นแบบอย่างที่ดีให้คนไทย ได้ดำเนินชีวิตบนทางสายกลาง มีความขยันหมั่นเพียรในการประกอบสัมมาชีพ รู้จักใช้ทรัพยากรที่มีอยู่ให้เกิดประโยชน์สูงสุด รักการประมาณตน และดำรงชีวิตอย่างรู้จัก คิด อยู่ ใช้ กิน อย่างพอเพียง

พระราชดำรัสที่เกี่ยวกับเศรษฐกิจพอเพียง

“...เศรษฐกิจศาสตร์เป็นวิชาของเศรษฐกิจ การที่ต้องใช้รถไถต้องไปซื้อ เราต้องใช้ต้องหาเงินมาสำหรับซื้อน้ำมันสำหรับรถไถ เวลารถไถเก่าเราต้องยิ่งซ่อมแซม แต่เวลาใช้นั้นเราก็ต้องบ่อน้ำมันให้เป็นอาหาร เสร็จแล้วมันคายคว้น คว้นเราสาดเข้าไปแล้วก็ปวดหัว ส่วนควายเวลาเราใช้เราก็ต้องบ่อนอาหาร ต้องให้หญ้าให้อาหารมันกิน แต่วามันคายออกมา ที่มันคายออกมาก็เป็นปุ๋ย แล้วก็ใช้ได้สำหรับให้ที่ดินของเราไม่เสีย...”

พระราชดำรัส เนื่องในพระราชพิธีพืชมงคลจรดพระนังคัลแรกนาขวัญ

ณ ศาลาดุสิดาลัย วันที่ ๙ พฤษภาคม ๒๕๒๙

“...การจะเป็นเช่นนั้นไม่สำคัญ สำคัญอยู่ที่เรามีเศรษฐกิจแบบพอมีพอกิน แบบพอมีพอกินนั้นหมายความว่า อุ่มชูตัวเองได้ ให้มีพอเพียงกับตนเอง ความพอเพียงนี้ไม่ได้หมายความว่าทุกครอบครัวจะต้องผลิตอาหารของตัวเอง จะต้องทอผ้าใส่เอง อย่างนั้นมันเกินไป แต่ว่าในหมู่บ้านหรือในอำเภอ จะต้องมีความพอเพียงพอสมควร บางสิ่งบางอย่างผลิตได้มากกว่าความต้องการก็ขายได้ แต่ขายในที่ไม่ห่างไกลเท่าไร ไม่ต้องเสียค่าขนส่งมากนัก...”

พระราชดำรัส เนื่องในโอกาสวันเฉลิมพระชนมพรรษา

ณ ศาลาดุสิดาลัย วันที่ ๔ ธันวาคม ๒๕๓๙.

“...พอเพียง มีความหมายกว้างขวางยิ่งกว่านี้อีก คือคำว่าพอ ก็พอเพียงนี้ก็พอแค่นั้นเอง คนเราถ้าพอในความต้องการก็มีความโลภน้อย เมื่อมีความโลภน้อยก็เบียดเบียนคนอื่นน้อย ถ้าประเทศใดมีความคิดอันนี้ มีความคิดว่าทำอะไรต้องพอเพียง หมายความว่าพอประมาณ ซื่อตรง ไม่โลภอย่างมาก คนเราก็อยู่เป็นสุข พอเพียงนี้อาจจะมี มีมากอาจจะมีของหรูหราก็ได้ แต่ว่าต้องไม่ไปเบียดเบียนคนอื่น...”

พระราชดำรัส เนื่องในโอกาสวันเฉลิมพระชนมพรรษา

ณ ศาลาดุสิดาลัย วันที่ ๔ ธันวาคม ๒๕๔๑

“...ไฟฟ้าถ้ามีความจำเป็น หากมีเศรษฐกิจพอเพียงแบบไม่เต็มที่ เรามีเครื่องปั่นไฟก็ใช้ปั่นไฟ หรือถ้าขั้นโบราณกว่า มีดก็จุดเทียน คือมีทางที่จะแก้ปัญหาเสมอ ฉะนั้นเศรษฐกิจพอเพียงก็มีเป็นขั้นๆ แต่จะบอกว่าเศรษฐกิจพอเพียงนี้ ให้พอเพียงเฉพาะตัวเองร้อยเปอร์เซ็นต์นี่เป็นสิ่งทำไม่ได้ จะต้องมีการแลกเปลี่ยน ต้องมีการช่วยกัน ถ้ามีการช่วยกัน แลกเปลี่ยนกัน ก็ไม่ใช่พอเพียงแล้ว แต่ว่าพอเพียงในทฤษฎีในหลวงนี้ คือให้สามารถที่จะดำเนินงานได้...”

พระราชดำรัส เนื่องในโอกาสวันเฉลิมพระชนมพรรษา

ณ ศาลาดุสิดาลัย วันที่ ๒๓ ธันวาคม ๒๕๔๒

ที่มา: [มูลนิธิชัยพัฒนา](#)

บทที่ 8 การสร้างกระดานข่าว

การสร้างกระดานข่าว เป็นส่วนหนึ่งของการสร้างระบบสารสนเทศบนระบบเครือข่าย เพราะสามารถรับความคิดเห็นของลูกค้า การสื่อสารข้อมูลต่าง ๆ การตอบคำถามของลูกค้า การติดต่อสัมพันธ์งานต่าง ๆ ซึ่งเป็นเครื่องมือที่ได้รับการตอบรับอย่างดี

ฐานข้อมูลกระดานข่าว

กระดานข่าวมีการเก็บคำถามคำตอบ เพราะฉะนั้น ฐานข้อมูลของการดานข่าวมี 2 ส่วน คือ กระดานข่าวคำถาม และ กระดานข่าวคำตอบ

ฐานข้อมูลกระดานข่าวคำถาม

	Field	Type	Collation	Attributes	Null	Default
☐	<u>q_id</u>	int(11)			No	
☐	q_topic	mediumtext	tis620_thai_ci		No	
☐	q_detail	mediumtext	tis620_thai_ci		No	
☐	q_name	varchar(50)	tis620_thai_ci		No	
☐	q_date	datetime			No	0000-00-00 00:00:00
☐	q_reply	int(11)			No	0
☐	q_ip	varchar(15)	tis620_thai_ci		No	

รูปที่ 8-1 : แสดงรายละเอียดตาราง webboard_question

ฐานข้อมูลกระดานข่าวคำตอบ

	Field	Type	Collation	Attributes	Null	Default
☐	<u>a_id</u>	int(11)			No	
☐	a_detail	mediumtext	tis620_thai_ci		No	
☐	a_name	mediumtext	tis620_thai_ci		No	
☐	a_date	datetime			No	0000-00-00 00:00:00
☐	a_ip	varchar(15)	tis620_thai_ci		No	
☐	q_id	int(11)			No	0

รูปที่ 8-2 : แสดงรายละเอียดตาราง webboard_answer

ฐานข้อมูลกระดานข่าวคำตอบจะเชื่อมกับฐานข้อมูลคำถามที่ ฟิลด์ q_id ว่าคำตอบนี้ เป็นการตอบคำถามไหน

ตัวอย่างที่ 8-1 ไฟล์ INDEX.PHP

```

1  <?
2  error_reporting(E_ALL ^ E_NOTICE);
3  $conn = mysql_connect("localhost","root","1234") or die ("Access Host denied.");
4  mysql_select_db("MIS",$conn) or die ("Access Database denied.");
5  #———— check bad word —————
6  function check($word)
7  {
8      $bad = Array("bad","damn");
9      for ($i=0;$i<SizeOf($bad); $i++)
10     {
11         $word = eregi_replace ($bad[$i],"<FONT COLOR=red>***</FONT>",$word);
12     }
13     return $word;
14 }
15 #———— test —————
16 //echo "TEST :: ".check("test damn")."<br>";
17 #=====แบ่งหน้า=====
18 #————1 ทำจำนวน Record ทั้งหมด —————
19 $sql = "select * from webboard_question";
20 $result = mysql_query($sql) or die(mysql_error());
21 $totalrows=mysql_num_rows($result);
22 #———— 2 กำหนดจำนวน Record ใน 1 หน้า ————
23 $pagesize = 5;
24 #———— 3 ทำจำนวนหน้าทั้งหมดที่แบ่งได้ —————
25 $totalpages = (int)($totalrows / $pagesize);
26 if ($totalrows % $pagesize != 0) $totalpages++;
27 #—————
28 #————4 หาจุด Start ในแต่ละหน้า
29 if (isset($_GET['page']))
30     $page = 1;
31 else
32     $page = $_GET['page'];
33 $start = ($page * $pagesize) - $pagesize;
34 $sql = "select * from webboard_question order by q_id desc limit $start,$pagesize";
35 $result = mysql_query($sql) or die(mysql_error());
36 ?>
37 <table border="1" align="center" cellpadding="5" cellspacing="0" bordercolor="#FFFFFF" bgcolor="#99CC00" >
38 <TR>
39     <Th colspan=4>Webboard :: <A HREF="newtopic.html">New TOPIC</A></Th>
40 </TR>
41 <TR>
42     <Th>Topic</Th>
43     <Th>Name</Th>
44     <Th>Date</Th>
45     <Th>Reply</Th>
46 </TR>

```

```

47 <? while ($row = mysql_fetch_array($result)) { ?>
48 <TR>
49   <TD><A HREF="showtopic.php?id=<?=$row['q_id']?>">
50   <? echo check(Substr($row[q_topic],0,10)) ;
51     if (strlen(check($row[q_topic])) > 10) echo "...";
52   ?></A> </TD>
53   <TD><?=$row['q_name']?></TD>
54   <TD><?=$row['q_date']?></TD>
55   <TD>
56   [<?=$row['q_reply']?>]</TD>
57 </TR>
58 <? } ?>
59 <tr>
60 <th colspan="4"> Pages ::
61 <?
62   if ($page > 1)
63   { ?>
64     &nbsp;<A HREF="index.php?page=<?=$page-1?>">Back</A>&nbsp;
65     <?
66   }
67 for ($i=1; $i<=$totalpages ; $i++)
68 {
69 //echo "<A HREF=\"index.php?page=$i\">$i</A>"
70   if ($i==$page)
71     echo $i; // หน้าปัจจุบัน ไม่แสดง link
72   else
73   {
74   ?>
75   &nbsp;<A HREF="index.php?page=<?=$i?>"><?=$i?></A>&nbsp;
76   <? }
77   }
78   if ($page < $totalpages)
79   {
80   ?>
81     &nbsp;<A HREF="index.php?page=<?=$page+1?>">Next</A>&nbsp;
82     <?
83   }
84   ?>
85 </th>
86 </tr>
87 </TABLE>

```

ผลลัพธ์

Webboard :: New TOPIC			
Topic	Name	Date	Reply
ob_start	นภัสศรีณย์ ชัชวาลานนท์	2009-11-18 15:51:02	[0]
????????????...	????????????? ??????????????	2009-11-18 15:46:21	[0]
Pages :: 1			

รูปที่ 8-3 : แสดงผลลัพธ์การแสดงผลภาษาไทย

อธิบายรหัสคำสั่ง

บรรทัดที่ 1-4 เป็นคำสั่งที่ใช้ในการติดต่อฐานข้อมูล

บรรทัดที่ 5-14 เป็นคำสั่งเพื่อแทนที่คำหยาบ โดยคำหยาบกำหนดไว้ในตัวแปรชื่อ bad โดยถ้ามีข้อความใดที่ตรงกับที่กำหนดในตัวแปร bad โปรแกรมจะแสดงข้อความ *** แทนข้อความหยาบ

บรรทัดที่ 19 เป็นคำสั่ง SQL เพื่อดึงข้อมูลของคำถามทั้งหมดจากตาราง Webboard_question

บรรทัดที่ 20 เป็นการรันคำสั่ง SQL

บรรทัดที่ 21 เป็นคำสั่งดึงจำนวนแถวออกมาว่าได้กี่แถวจากการรันคำสั่ง SQL

บรรทัดที่ 23 เป็นคำสั่งเพื่อกำหนดจำนวนระยะเบี่ยนในแต่ละหน้า

บรรทัดที่ 25 เป็นการคำนวณหาจำนวนหน้าของเอกสาร

บรรทัดที่ 26 เป็นการตรวจสอบหน้าที่เหลือว่ามีเศษหรือไม่ ถ้ามีเศษ ให้เพิ่มจำนวนหน้าอีก 1 หน้า

บรรทัดที่ 29-32 เป็นคำสั่งเพื่อตรวจสอบหน้าปัจจุบันว่าตอนนี้ อยู่ที่หน้าไหน ถ้าเป็นครั้งแรกให้กำหนดเป็นหน้าแรก

บรรทัดที่ 33 เป็นการคำนวณหาระเบี่ยนเริ่มต้นของหน้าเอกสาร หน้านี้

บรรทัดที่ 34 เป็นคำสั่ง SQL เพื่อดึงข้อมูลของคำถามโดยเรียงตามเลขมากโดยเริ่มจากระเบี่ยนที่ตั้งค่าเริ่มต้นไว้ โดยนำมาแสดง เท่ากับ จำนวนระยะเบี่ยน ที่กำหนดไว้

บรรทัดที่ 35 เป็นการรันคำสั่ง SQL

บรรทัดที่ 49 เป็นคำสั่งกำหนด link ให้กับคำถาม โดยกำหนดให้ทำงานที่ ShowTopic โดยส่งรหัสคำถาม ไปกับ Link ด้วย โดยผ่าน Method GET

บรรทัดที่ 50 เป็นคำสั่ง แสดงคำถาม โดยมีการตรวจสอบ คำหยาบก่อน

บรรทัดที่ 62-66 เป็นคำสั่ง แสดงคำว่า Back โดยมี Link ไปยังหน้า ก่อนหน้า โดยแสดงเฉพาะหน้า 2 เป็นต้นไป

บรรทัดที่ 67-77 เป็นคำสั่ง แสดง หน้า 1 2 3 ไปจนถึงจำนวนหน้าทั้งหมด โดยมี link ทุกหน้า ยกเว้นหน้าปัจจุบัน

บรรทัดที่ 78-84 เป็นคำสั่ง แสดงคำว่า Next โดยมี Link ไปยังหน้าต่อไป โดยแสดงทุกครั้ง ยกเว้นเป็นหน้าสุดท้าย

การแก้ไขปัญหภาษาไทย

ผลลัพธ์ที่ได้จากการรันโปรแกรมถ้าข้อมูลเป็น ภาษาไทยในการแสดงและการเก็บข้อมูล ลงฐานข้อมูลต้องระบุรหัสภาษาที่ใช้ในการเก็บลงฐานข้อมูลด้วย เช่น ภาษาไทย ส่วนใหญ่ใช้ ภาษาไทย TIS-620 จะใช้คำสั่ง `mysql_query("SET CHARACTER SET tis620");` ดังนั้น ขั้นตอนการติดต่อฐานข้อมูลควรสร้างไฟล์ไว้ต่างหาก มิฉะนั้นจะต้องแก้ทุกไฟล์ที่เกี่ยวข้อง ควร แยกเก็บไฟล์ใน connect.php

ตัวอย่างที่ 8-2 ไฟล์ connect.php

```

1  <?
2  $conn = @mysql_connect("localhost","root","1234") or die("cannot connect DB Server");
3  $db = mysql_select_db("MIS") or die("cannot select DB");
▶4  mysql_query("SET CHARACTER SET tis620");
5  ?>
6

```

การสร้างคำถาม

การสร้างคำถาม หรือการตั้งกระทู้ ก็คือ การเพิ่มข้อมูลลงไปในฐานข้อมูลนั่นเอง คือมี การสร้างฟอร์มเพื่อรับข้อมูล แล้วส่งให้ไฟล์ PHP ประมวลผลเพื่อทำการ Insert ลงฐานข้อมูล ไฟล์ที่เกี่ยวข้องคือ การสร้างฟอร์มกรอกข้อมูล ชื่อ NewTopic.HTML และไฟล์ที่ใช้ในการเพิ่ม ข้อมูลลงฐานข้อมูล คือ NewTopic.PHP

ตัวอย่างที่ 8-3 ไฟล์ NewTopic.HTML

```

1 <FORM METHOD=POST ACTION="newtopic.php" enctype="MULTIPART/FORM-DATA">
2 <table border="1" align="center" cellpadding="5" cellspacing="0" bordercolor="#FFFFFF" bgcolor="#99CC00" >
3 <tr>
4   <th colspan="2">NEW TOPIC :: <a href="INDEX.PHP">HOME</a> </th>
5 </tr>
6 <TR>
7   <TD>Topic :: </TD>
8   <TD><INPUT TYPE="text" NAME="txtTopic" size="55"><br></TD>
9 </TR>
10 <TR>
11   <TD>Detail :: </TD>
12   <TD><TEXTAREA NAME="txtDetail" ROWS="10" COLS="55" ></TEXTAREA><br></TD>
13 </TR>
14 <TR>
15   <TD>Name :: </TD>
16   <TD><INPUT TYPE="text" NAME="txtName" size="55"><br></TD>
17 </TR>
18 <TR>
19   <TD>Picture :: </TD>
20   <TD><INPUT TYPE="file" NAME="txtFile" size="50"><br></TD>
21 </TR>
22 <TR>
23   <TD><INPUT TYPE="submit" value = "P O S T"></TD>
24   <TD><INPUT TYPE="reset" Value ="C L E A R"></TD>
25 </TR>
26 </TABLE>
27 </FORM>

```

ผลลัพธ์

NEW TOPIC :: HOME	
Topic ::	session_start() หมายถึง อะไร
Detail ::	session_start() หมายถึง อะไร
Name ::	ตฤณี ชัชวาลานนท์
Picture ::	D:\Mydoc\My Pictures\11636_001.jpg Browse...
POST CLEAR	

รูปที่ 8-4 : แสดงการตั้งคำถามใหม่

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นคำสั่งสร้างฟอร์มโดยมี type แบบ MultiPart คือส่งข้อมูลได้ทั้งแบบข้อความและไฟล์

บรรทัดที่ 2-27 เป็นคำสั่งเพื่อรับข้อมูล

ตัวอย่างที่ 8-4 ไฟล์ NewTopic.PHP

```

1  <?
2  error_reporting(E_ALL ^ E_NOTICE);
3  $conn = mysql_connect("localhost","root","1234") or die ("Access Host denied.");
4  mysql_select_db("MIS",$conn) or die ("Access Database denied.");
5  mysql_query("SET CHARACTER SET tis620");
6  if (empty($_FILES['txtFile']['name']))
7  {
8      #----- file -----
9      echo "File Name :: ".$_FILES['txtFile']['name']."<br>";
10     echo "File Size :: ".$_FILES['txtFile']['size']."<br>";
11     echo "File Type :: ".$_FILES['txtFile']['type']."<br>";
12     echo "File Temp_name :: ".$_FILES['txtFile']['tmp_name']."<br>";
13     #-----
14     if ($_FILES['txtFile']['size'] > (50*1024))
15     {
16         echo "<Script> alert('FILE size over flow') ; history.back(); </Script>";
17         exit;
18     }
19     if (Substr($_FILES['txtFile']['type'],0,5) != "image")
20     {
21         echo "<Script> alert('Only Image File') ; history.back(); </Script>";
22         exit;
23     }
24 }
25 $sql = "insert into webboard_question values
('".$_POST['txtTopic']."','".$_POST['txtDetail']."','".$_POST['txtName']."',now(),0,'".$_ENV['REMOTE_ADDR']."')";
26 $result = mysql_query($sql) or die(mysql_error());
27 if ($result)
28 {
29     echo "insert OK<br>";
30     if (empty($_FILES['txtFile']['name']))
31     {
32         $filename = mysql_insert_id();
33         if (copy($_FILES['txtFile']['tmp_name'], "Images/$filename.jpg"))
34             echo "Copy File OK";
35         else
36             echo "Copy File Fail";
37     }
38 }
39 else
40     echo "insert Fail";
41 mysql_close($conn);
42 echo "<Script>";
43 echo "alert ('THANK YOU'); ";
44 echo " window.location ='index.php'; ";
45 echo "</script>";
46 ?>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 5 เป็นคำสั่งเพื่อกำหนดรหัสภาษาไทย ในการเพิ่มข้อมูลลงในฐานข้อมูล ให้เป็นรหัสภาษาไทย TIS-620

บรรทัดที่ 6 เป็นการตรวจสอบว่ามีการใส่ชื่อรูปหรือไม่

บรรทัดที่ 8-12 เป็นคำสั่งเพื่อแสดงรายละเอียดของไฟล์ ซึ่งไม่แสดงก็ได้

บรรทัดที่ 14-18 เป็นคำสั่งเพื่อตรวจสอบว่าไฟล์ใหญ่เกิน 5 MB หรือไม่

ถ้าเกินจะแสดงข้อความ File Size Over Flow แล้วกลับไปให้คีย์ข้อมูลใหม่

บรรทัดที่ 19-24 เป็นคำสั่งเพื่อตรวจสอบว่าไฟล์เป็นไฟล์รูปภาพหรือไม่ ถ้าไม่ใช่

จะแสดงข้อความ Only Image File แล้วกลับไปให้คีย์ข้อมูลใหม่

บรรทัดที่ 25-26 เป็นคำสั่ง SQL เพื่อเพิ่มข้อมูล ลงฐานข้อมูล ตาราง webboard_question

บรรทัดที่ 27-38 เป็นคำสั่ง เพื่อตรวจสอบว่าถ้าเพิ่มข้อมูลสำเร็จให้ตรวจสอบว่ามีไฟล์

ให้ทำการ COPY ไฟล์รูปภาพ ขึ้น Host โดยตั้งชื่อเป็น รหัสคำถามและนามสกุล jpg

บรรทัดที่ 44 เป็นคำสั่งเพื่อ ไปทำงานที่ ไฟล์ index.php

ผลลัพธ์

```
File Name :: 11636_001.jpg
File Size :: 4642
File Type :: image/jpeg
File Temp_name :: C:\WINDOWS\TEMP\php6D.tmp
insert OK
Copy File OK
```



รูปที่ 8-5 : แสดงผลลัพธ์การตั้งคำถามใหม่

Webboard :: <u>New TOPIC</u>			
Topic	Name	Date	Reply
session st...	ดร.ณิ ชัชวาลานนท์	2009-11-18 16:31:06	[0]
ob_start	นภัสศรีณย์ ชัชวาลานนท์	2009-11-18 15:51:02	[0]
Pages :: 1			

รูปที่ 8-6 : แสดงผลลัพธ์การแสดงผลหัวข้อคำถาม

การแสดงคำถาม

การแสดงคำถาม มีลักษณะเหมือนกับการแสดงข้อมูลจากฐานข้อมูลทั่วไป โดยจะต้องแสดงคำถาม แล้วต้องแสดงคำตอบด้วย ถ้ามี และต้องมีการแสดงรูปภาพด้วย ถ้ามีรูปของคำถามนั้น ๆ และรอรับคำตอบอีกด้วย

ตัวอย่างที่ 8-5 ไฟล์ ShowQ.PHP

```

1  <?
2  error_reporting(E_ALL ^ E_NOTICE);
3  $conn = mysql_connect("localhost","root","1234") or die ("Access Host denied.");
4  mysql_select_db("MIS",$conn) or die ("Access Database denied.");
5  mysql_query("SET CHARACTER SET tis620");
6  $q_id = $_GET['id'];
7  $sql = "select * from webboard_question where q_id = '$q_id' ";
8  $result = mysql_query($sql) or die(mysql_error());
9  $rs = mysql_fetch_array($result);
10 ?>
11 <table border="1" align="center" cellpadding="5" cellspacing="0" bordercolor="#FFFFFF" bgcolor="#99CC00" width="400">
12 <TR>
13   <TD colspan=2><?if (file_exists("images/$q_id.jpg"))
14       echo "<center><img src='images/$q_id.jpg'><center>";
15   ?><br>
16   <h2><?=$rs['q_topic']?></h2></TD>
17 </TR>
18 <TR>
19   <TD>Detail</TD>
20   <TD><?=$rs['q_detail']?></TD>
21 </TR>
22 <TR>
23   <TD>Name</TD>
24   <TD><?=$rs['q_name']?></TD>
25 </TR>
26 <TR>
27   <TD>Date</TD>
28   <TD><?=$rs['q_date']?></TD>
29 </TR>
30 </TABLE>

31 <!-- begin loop show answer ----->
32 <?
33 $sql= "select * from webboard_answer where q_id = '$q_id' ";
34 $result = mysql_query($sql) or die(mysql_error());
35 $x=1;
36 while ($row = mysql_fetch_array($result)) {
37 ?>
38 <center>
39 <TABLE border=1 cellpadding="5" cellspacing="0" bordercolor="#CCCCFF" bgcolor="#CC99FF" align="center">
40   <tr>
41     <td>

```

```

42  <?
43  if (file_exists("images/A_{$row[0].jpg}"))
44  {
45      echo "<center><img src = \"images/A_{$row[0].jpg}\"></center>";
46  }
47  ?> <BR>ความเห็นคนที่ <?=$x++?><br>
48      Detail :: <?=$row['a_detail']?> <br>
49      Name :: <?=$row['a_name']?><br>
50      Date :: <?=$row['a_Date']?><br>
51  </td>
52 </TR>
53 </TABLE> <br>
54 <?} ?>
55 <!-- end loop show snawer ----->
56 <FORM METHOD=POST ACTION="answer.php?id=<?=$q_id?>" enctype="MULTIPART/FORM-DATA">
57 <INPUT TYPE="hidden" name = "q_id" value="<?=$GET['q_id']?>">
58 <TABLE align="center">
59 <TR>
60 <TD>Detail :: <TEXTAREA NAME="txtDetail" ROWS="3" COLS="50"></TEXTAREA><br></TD>
61 </tr>
62 <tr>
63 <td>Name :: <INPUT TYPE="text" NAME="txtName"><br></td>
64 </tr>
65 <tr>
66 <td>Picture :: <INPUT TYPE="file" NAME="txtPic"><br></td>
67 </TR>
68 <tr>
69 <td><INPUT TYPE="submit" value = "ส่งความคิดเห็น"></td>
70 </TR>
71 </TABLE>
72 </FORM>
73 <? mysql_close($conn) ?>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1-10 เป็นคำสั่งที่ดึงข้อมูลคำถามที่ต้องการมาจากรฐานข้อมูล

บรรทัดที่ 13-14 คำสั่งเพื่อตรวจสอบว่ามีรูปภาพของคำถามนี้ หรือไม่ ถ้ามีให้แสดงรูปภาพของคำถามด้วย

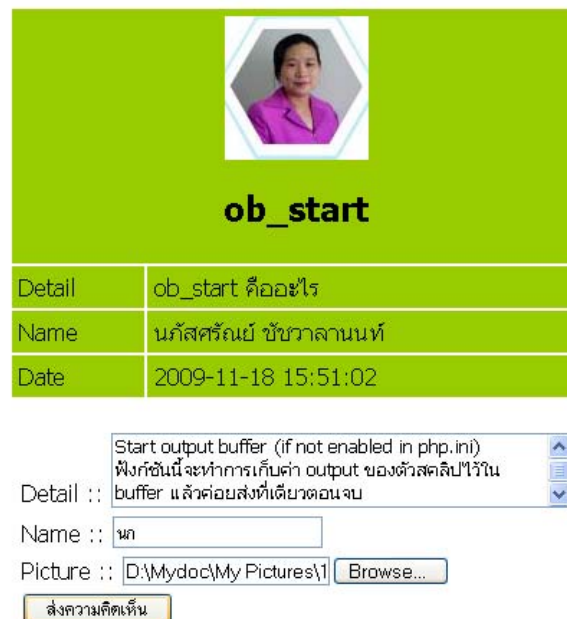
บรรทัดที่ 33-36 คำสั่งดึงข้อมูลของคำตอบของคำถามนี้

บรรทัดที่ 45 การแสดงรูปภาพ ของคำตอบ

บรรทัดที่ 47-52 การแสดงข้อมูลคำตอบ

บรรทัดที่ 56 การสร้างฟอร์มเพื่อรับข้อมูลคำตอบใหม่ โดยมี link ไปที่ไฟล์ answer.php โดยแนบ รหัสคำถาม ตามไปด้วย สร้างฟอร์ม แบบ MultiPart คือส่งข้อมูลได้ทั้งแบบข้อความและไฟล์

ผลลัพธ์



The screenshot shows a web application interface with a green header. At the top, there is a profile picture of a woman in a pink shirt. Below the picture, the name 'ob_start' is displayed in bold. Underneath the name is a table with three rows: 'Detail' with the value 'ob_start คืออะไร', 'Name' with the value 'นภัสศรีณย์ ชัชวาลานนท์', and 'Date' with the value '2009-11-18 15:51:02'. Below the table, there is a text area with the text 'Start output buffer (if not enabled in php.ini) ฟังก์ชันนี้จะทำการเก็บค่า output ของตัวสคริปต์ไว้ใน buffer แล้วค่อยส่งทีเดียวตอนจบ'. Below the text area, there is a 'Name' field with the value 'นภ' and a 'Picture' field with the value 'D:\Mydoc\My Pictures\1'. There is a 'Browse...' button next to the 'Picture' field. At the bottom, there is a 'ส่งความคิดเห็น' button.

รูปที่ 8-7 : แสดงการตอบคำถาม

การตอบคำถาม

การตอบคำถามก็มีลักษณะการสร้างคำถาม คือ การเพิ่มข้อมูลลงฐานข้อมูล โดยมีรายละเอียดเพิ่มเติม คือ คำถามนี้ เป็นคำตอบของคำถามไหน และมีรูปภาพตามมาด้วย ต้องมีการ Upload ขึ้นเครื่องแม่ข่าย และเมื่อมีการตอบคำถาม ต้องมีการ Update จำนวนคำตอบเพิ่มขึ้นอีกหนึ่งด้วย

ตัวอย่างที่ 8-6 ไฟล์ Answer.PHP

```

1 <?
2 error_reporting(E_ALL ^ E_NOTICE);
3 $conn = mysql_connect("localhost","root","1234") or die ("Access Host denied.");
4 mysql_select_db("MIS",$conn) or die ("Access Database denied.");
5 mysql_query("SET CHARACTER SET tis620");
6 $q_id = $_GET['id'];
7 if (empty($_FILES['txtPic']['name']))
8 {
9     if ($_FILES['txtPic']['size'] > 51200)
10     {
11         echo "<script>alert('Over File size'); history.back(); </script>";
12         exit;
13     }
14     if (SubStr($_FILES['txtPic']['type'],0,5) != "image")
15     {
16         echo "<script>alert('Not Picture Type'); history.back(); </script>";
17         exit;
18     }
19 }

```

```

20 $sql = "insert into webboard_answer values('".$_POST['txtDetail']."','".$_POST['txtName']."'.NOW(),'{$_ENV[REMOTE_ADDR]}','$q_id')";
21 //echo $sql;

22 $result = mysql_query($sql) or die(mysql_error());
23 if ($result)
24 {
25     echo "insert OK<BR>";
26
27     //echo $_FILES['txtPic']['name'];
28     if (!empty($_FILES['txtPic']['name']))
29     {
30         $filename = "A_".mysql_insert_id().".jpg";
31         echo "FileName : $filename <br>";
32         if (copy($_FILES['txtPic']['tmp_name'], "images/$filename"))
33             echo "copy file picture ok<br>";
34         else echo "copy File Fail";
35     }
36     // ----- update q_reply -----
37     $sql = "update webboard_question set q_reply = q_reply + 1 where q_id = '$q_id'";
38     $result = mysql_query($sql) or die(mysql_error());
39     if ($result) echo "update q_reply OK<BR>"; else echo "update fail";
40     // -----
41 }
42 else echo "inser Fail";
43 mysql_close($conn);
44 echo "<Script>";
45 echo "alert ('THANK YOU'); ";
46 echo "window.location ='index.php'; ";
47 echo "</script>";
48 ?>

```

ผลลัพธ์

insert OK
 FileName : A_42.jpg
 copy file picture ok
 update q_reply OK



รูปที่ 8-8 : แสดงผลลัพธ์การตอบคำถาม

อธิบายรหัสคำสั่ง

บรรทัดที่ 1-4 เป็นคำสั่งเพื่อติดต่อเครื่องแม่ข่าย ฐานข้อมูล

บรรทัดที่ 5 เป็นคำสั่งเพื่อกำหนดรหัสภาษาไทย ในการเพิ่มข้อมูลลงในฐานข้อมูลให้เป็นรหัสภาษาไทย TIS-620

บรรทัดที่ 6 เป็นการรับค่า รหัสคำถามมาด้วย Method GET

บรรทัดที่ 7 เป็นการตรวจสอบว่ามีการใส่ชื่อรูปหรือไม่

บรรทัดที่ 9-13 เป็นคำสั่งเพื่อตรวจสอบว่าไฟล์ใหญ่เกิน 5 MB หรือไม่ ถ้าเกินจะแสดงข้อความ File Size Over Flow แล้วกลับไปให้คีย์ข้อมูลใหม่

บรรทัดที่ 14-19 เป็นคำสั่งเพื่อตรวจสอบว่าไฟล์เป็นไฟล์รูปภาพหรือไม่ ถ้าไม่ใช่จะแสดงข้อความ Only Image File แล้วกลับไปให้คีย์ข้อมูลใหม่

บรรทัดที่ 20 เป็นคำสั่ง SQL เพื่อเพิ่มข้อมูล ลงฐานข้อมูล ตาราง webboard_answer

บรรทัดที่ 23-33 เป็นคำสั่ง เพื่อตรวจสอบว่าถ้าเพิ่มข้อมูลสำเร็จให้ตรวจสอบว่ามีไฟล์ให้ทำการ COPY ไฟล์รูปภาพ ขึ้น Host โดยตั้งชื่อเป็น รหัสคำตอบและนามสกุล jpg

บรรทัดที่ 36-39 เป็นคำสั่งเพื่อ Update จำนวนคำตอบเพิ่มอีกหนึ่ง

ผลลัพธ์

Webboard :: New TOPIC			
Topic	Name	Date	Reply
session st...	ดร.ณัฏฐ์ ชัชวาลานนท์	2009-11-18 16:31:06	[0]
ob_start	นภัสศรีณย์ ชัชวาลานนท์	2009-11-18 15:51:02	[1]
Pages :: 1			

รูปที่ 8-9 : แสดงผลลัพธ์การแสดงผลจำนวนคำตอบของคำถาม

ผลลัพธ์

	
ob_start	
Detail	ob_start คืออะไร
Name	นภัสศรีณย์ ชัชวาลานนท์
Date	2009-11-18 15:51:02

	
ความเห็นคนที่ 1 Detail :: Start output buffer (if not enabled in php.ini) ฟังก์ชันนี้จะทำการเก็บค่า output ของตัวสคริปต์ไว้ใน buffer แล้วค่อยส่งทีเดียวตอนจบ Name :: นก Date :: 2009-11-18 16:52:54	

Detail ::	
Name ::	
Picture ::	Browse...
ส่งความคิดเห็น	

รูปที่ 8-10 : แสดงความคิดเห็นและคำตอบ

พระราชดำรัสเกี่ยวกับการศึกษา ของพระบาทสมเด็จพระเจ้าอยู่หัว

"การแบ่งการศึกษาเป็นสองอย่าง คือการศึกษาวิชาการอย่างหนึ่ง วิชาการนั้นจะเป็นประโยชน์แก่ตัวเองและแก่บ้านเมือง ถ้ามาใช้ต่อไปเมื่อสำเร็จการศึกษาแล้ว อีกอย่างหนึ่ง ขั้นที่สองก็คือความรู้ที่จะเรียกว่าวัฒนธรรม คือรู้ในการวางตัว ประพฤติและคิด วิธีคิด วิธีที่จะใช้สมองมาทำเป็นประโยชน์แก่ตัว สิ่งที่เป็นธรรมหมายถึงวิธีประพฤติปฏิบัติ คนที่ศึกษาในทางวิชาการและศึกษาในทางธรรมก็ต้องมีปัญญา แต่ผู้มีความรู้ในทางวิชาการทางเดียวและไม่ใช้ความรู้ในทางธรรมจะนับว่าเป็นปัญญาชนมิได้"

(๑๘ ธันวาคม ๒๕๑๗)

บทที่ 9 การทำแบบสำรวจ

การทำแบบสำรวจ หรือ การทำโพล ต้องมีการสร้างคำถาม และคำตอบของคำถาม เพื่อให้รู้ว่าคำตอบนี้ เป็นคำตอบของคำถามนั้น โดยใช้ฟิลด์ `q_id` เป็นตัวเชื่อมระหว่างตาราง โดยมีตาราง 2 ตาราง คือ `poll_question` และ `poll_answer`

ฐานข้อมูล `poll_question`

ฐานข้อมูลประกอบด้วย ฟิลด์ 2 ฟิลด์ คือ รหัสคำถาม และ รายละเอียดคำถาม โดยรหัสคำถาม เป็นเลขจำนวนเต็ม และกำหนดเป็น `Auto_increment` คือ รหัสคำถามให้รันอัตโนมัติ และ รายละเอียดคำถาม เป็น ตัวอักษร มีความกว้าง 255 ตัวอักษร และเก็บด้วยรหัสภาษาไทย TIS-620

	Field	Type	Collation
☐	<u>q_id</u>	int(11)	
☐	q_detail	varchar(255)	tis620_thai_ci

รูปที่ 9-1 : แสดงรายละเอียดตาราง `poll_question`

ฐานข้อมูล `poll_answer`

ฐานข้อมูลประกอบด้วย ฟิลด์ 4 ฟิลด์ คือ รหัสคำตอบ และ รายละเอียดคำตอบ โดยรหัสคำตอบ เป็นเลขจำนวนเต็ม และกำหนดเป็น `Auto_increment` คือ รหัสคำตอบให้รันอัตโนมัติ และ รายละเอียดคำตอบ เป็น ตัวอักษร มีความกว้าง 255 ตัวอักษร และเก็บด้วยรหัสภาษาไทย TIS-620 และ จำนวนคะแนนที่โหวต เป็นจำนวนตัวเลขจำนวนเต็ม โดยจะเพิ่มขึ้นเรื่อย ๆ ตามจำนวนคนที่โหวต และ รหัสอ้างอิงกับคำถาม ว่าคำตอบนี้ เป็นของคำถามไหน

	Field	Type	Collation
☐	<u>a_id</u>	int(11)	
☐	a_detail	varchar(255)	tis620_thai_ci
☐	a_score	int(11)	
☐	q_id	int(11)	

รูปที่ 9-2 : แสดงรายละเอียดตาราง `poll_answer`

การสร้างแบบสำรวจ เริ่มจากไฟล์เพิ่มคำถาม โดยสามารถใส่จำนวนคำตอบได้ว่ามีกี่คำตอบ คือไฟล์ `AddForm.php` แล้วส่งไปให้ เพิ่มข้อมูลในฐานข้อมูล โดยเพิ่มข้อมูล ในตาราง

Poll_question และ ตาราง Poll_answer เมื่อเพิ่มคำถาม และคำตอบลงในฐานข้อมูลแล้ว จะแสดงข้อมูล โดยใช้ไฟล์ Index.php คือแสดงคำถาม และ คำตอบ พร้อมทั้งรอรับค่าการโหวต แล้วส่งคำตอบให้ไฟล์ Vote.php เพื่อทำการ Update จำนวนคำตอบให้เพิ่มขึ้นอีก 1 แล้วแสดงผลโหวตเป็นเปอร์เซ็นต์ โดยใช้ไฟล์ View.php

ตัวอย่างที่ 9-1 ไฟล์ Addform.php

```

1  <form action="" method="post">
2  <center> จำนวนคำตอบ
3  <input type="text" name="num">
4  <input type="submit" name="Submit3" value="Submit">
5  </form>
6  <form name="form1" method="post" action="addsave.php">
7  <table width="400" border="1" align="center" cellpadding="0" cellspacing="1">
8  <tr>
9  <td colspan="2"><strong>AddPoll</strong></td>
10 </tr>
11 <tr>
12 <td width="70"><strong>question</strong></td>
13 <td width="321"><input type="text" name="question"></td>
14 </tr>
15 <?
16 if(empty($_POST['num'])){ $num=$_POST['num']; }else{ $num=3; }
17 for($i=1;$i<=$num;$i++)
18 { ?>
19 <tr>
20 <td><strong>answer</strong></td>
21 <td><input type="text" name="answer[]"></td>
22 </tr>
23 <? } ?>
24 <tr>
25 <td colspan="2" align="center">
26 <input type="submit" name="Submit" value="Submit">
27 <input type="reset" name="Submit2" value="Reset"></td>
28 </tr>
29 </table>
30 </form>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นคำสั่ง <form> กำหนด method เป็น Post และ ไม่ส่งไฟล์ไปไหน เพราะจะนั้นส่งไปไฟล์เดิม

บรรทัดที่ 2 เป็นคำสั่ง กำหนดข้อมูลตรงกลาง แสดงข้อความ จำนวนคำตอบ

บรรทัดที่ 3 เป็นคำสั่ง รับจำนวนคำตอบ ของคำถาม ถ้าไม่ต้องการ 3 คำตอบ

บรรทัดที่ 4 เป็นคำสั่ง สร้างปุ่ม เพื่อส่งข้อมูล

บรรทัดที่ 5 เป็นคำสั่ง ปิดฟอร์มส่วนแรก

บรรทัดที่ 6 เป็นคำสั่ง สร้างฟอร์ม กำหนดชื่อ form1 และส่งไปทำงานที่ AdSave.php
 บรรทัดที่ 7 เป็นคำสั่ง สร้างตารางให้มีความกว้าง 400 มีเส้นขอบเท่ากับ 1
 บรรทัดที่ 16 เป็นคำสั่งตรวจสอบตัวแปร num ว่ามีหรือไม่ ถ้ามีแสดงว่า มีการส่งค่าจากฟอร์ม
 ตัวเองมา ถ้าไม่มี ให้มีค่าเท่ากับ 3
 บรรทัดที่ 17-23 เป็นคำสั่ง สร้างคำตอบให้ตรงกับ จำนวนคำถามที่ต้องการ โดยตัวแปร
 answer เป็นตัวแปรอาเรย์ สามารถเพิ่มได้ตามจำนวน ตัวแปร num

ผลลัพธ์

รูปที่ 8-3 : แสดงการรับจำนวนคำตอบแล้วสร้างคำตอบ

ตัวอย่างที่ 9-2 ไฟล์ Connect.php

```

1 <?
2 $conn = @mysql_connect("localhost","root","1234") or die("cannot connect DB");
3 $db = mysql_select_db("MIS") or die("cannot select DB");
▶4 mysql_query("SET CHARACTER SET tis620");
5 ?>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เครื่องหมาย <? หมายถึง เปิดคำสั่ง PHP
 บรรทัดที่ 2 เป็นคำสั่ง เพื่อเปิดการเชื่อมต่อกับเครื่องแม่ข่าย โดยใช้เครื่องหมาย @ เพื่อป้องกัน
 ไม่ให้แสดงข้อผิดพลาด กรณีที่เป็นเพียงคำเตือน warning
 บรรทัดที่ 3 เป็นคำสั่งเพื่อติดต่อฐานข้อมูล ชื่อ MIS
 บรรทัดที่ 4 เป็นคำสั่งกำหนด รหัสภาษาไทย ในการติดต่อฐานข้อมูล ไม่ว่าจะอ่านหรือเขียนให้
 เป็น รหัสภาษาไทย TIS-620
 บรรทัดที่ 5 เป็นคำสั่ง ปิดการเขียนคำสั่ง ของ PHP

ตัวอย่างที่ 9-3 ไฟล์ AddSave.php

```

1  <?
2  include("connect.php");
3  $question = $_POST['question'];
4  $answer    = $_POST['answer'];
5  // Add question
6  $sqlQ = "insert into poll_question values('','$question')";
7  mysql_query($sqlQ) or die("error=$sqlQ");
8  // Get last id
9  $q_id = mysql_insert_id();
10 | // Add question
11 for($i=0;$i<sizeof($answer);$i++)
12 {
13     if(!empty($answer[$i]))
14     {
15         //echo "[$i] => ".$answer[$i]."<br>";
16         $sqlA = "insert into poll_answer values('','$answer[$i]','0','$q_id')";
17         mysql_query($sqlA) or die("error=$sqlA");
18     }
19 }
20 echo "<script>alert('Complete');window.location='index.php';</script>";
21 ?>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เครื่องหมาย <? หมายถึง เปิดคำสั่ง PHP

บรรทัดที่ 2 เป็นคำสั่งเพื่อดึงรหัสคำสั่งจากไฟล์ Connect.php มาทำงาน

บรรทัดที่ 3-4 เป็นคำสั่งเพื่อดึงตัวแปร question และ answer จากฟอร์ม ผ่าน method POST

บรรทัดที่ 6 เป็นคำสั่งเพื่อเพิ่มคำถามที่รับมาจากฟอร์ม ผ่านตัวแปร question เพิ่มลงฐานข้อมูล ในตาราง poll_question

บรรทัดที่ 7 เป็นคำสั่งเพื่อรันคำสั่ง SQL

บรรทัดที่ 8 เป็นคำสั่งหมายเหตุ ของโปรแกรม PHP

บรรทัดที่ 9 เป็นคำสั่งเพื่อดึงหมายเลข อัตโนมัติ ของคำถามว่าเป็นเลขอะไร ไว้ในตัวแปร q_id

บรรทัดที่ 11-19 เป็นคำสั่งเพิ่มคำตอบลงในตารางฐานข้อมูล โดยเพิ่มตามจำนวนของคำตอบ โดยใช้ loop for และมีการตรวจสอบการตอบคำตอบว่ามีหรือไม่ ถ้าไม่มีก็ไม่ต้องเพิ่มลงฐานข้อมูล และเพิ่มรหัสคำถามด้วย เป็นการเชื่อมโยงระหว่างตาราง

บรรทัดที่ 20 เป็นคำสั่งแสดงข้อความ Complete และไปทำงานที่ index.PHP ต่อไป

ผลลัพธ์



รูปที่ 9-3 : แสดงผลลัพธ์การเพิ่มข้อมูล

ตัวอย่างที่ 9-4 ไฟล์ Index.php

```

1 <meta http-equiv="Content-Type" content="text/html; charset=TIS-620" />
2 <body text="#6699FF">
3   <form name="form1" method="post" action="vote.php">
4     <table width="400" border="1" align="center" bordercolor="#99CCFF">
5       <? include("connect.php");
6       $sqlQ = "select * from poll_question order by q_id desc";
7       $queryQ = mysql_query($sqlQ) or die("error=$sqlQ");
8       $totalrows=mysql_num_rows($queryQ);
9       $rowQ = mysql_fetch_array($queryQ);
10      $q_id = $rowQ['q_id'];
11      $q_detail = $rowQ['q_detail'];  ?>
12      <tr><td colspan="2" bgcolor="#DDEEFF"><?=$q_detail?></td></tr>
13      <? $sqlA = "select * from poll_answer where q_id=$q_id";
14        $queryA = mysql_query($sqlA) or die("error=$sqlA");
15        $numA = mysql_num_rows($queryA);
16        for($i=1;$i<=$numA;$i++) | {
17          $rowA = mysql_fetch_array($queryA);
18        ?>
19      <tr>
20        <td width="37" bgcolor="#F4FAFF">
21          <input name="a_id" type="radio" value="<?=$rowA['a_id']?>"></td>
22          <td width="354" bgcolor="#F4FAFF"><?=$rowA['a_detail']?></td>
23        </tr>
24      <? } ?>
25      <tr>
26        <td colspan="2" align="center" bgcolor="#DDEEFF">
27          <input type="submit" name="Submit" value="Vote">
28          <input type="button" name="Submit2" value="View" onClick="window.location='vote.php';">
29        </td>
30      </tr>
31    </table>
32  </form>
33

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นคำสั่ง HTML เพื่อกำหนดการแสดงผลในหน้าเว็บเป็นภาษาไทย ด้วยรหัสภาษาไทย TIS-620

บรรทัดที่ 2 เป็นคำสั่ง HTML เพื่อเปิดกำหนดค่า สีอักษรเป็นสี

บรรทัดที่ 3 เป็นคำสั่ง HTML เพื่อกำหนดชื่อ ฟอรัม ชื่อ Form1 และส่งค่าไปให้ไฟล์ Vote.php เมื่อกดปุ่ม submit

บรรทัดที่ 6-10 เป็นคำสั่ง เพื่อดึงคำถามจากฐานข้อมูล คำถามสุดท้าย

บรรทัดที่ 12 เป็นคำสั่งเพื่อแสดงคำถาม ที่ดึงมาจากฐานข้อมูล

บรรทัดที่ 13-15 เป็นคำสั่ง เพื่อดึงคำตอบของคำถามนี้ จากฐานข้อมูล

บรรทัดที่ 16-24 เป็นคำสั่งเพื่อแสดงคำตอบของคำถามนี้ ที่ดึงมาจากฐานข้อมูล โดยวนลูป for

บรรทัดที่ 28 เป็นคำสั่ง เพื่อกำหนดว่า ปุ่ม View นี้ เป็นปุ่ม ถ้ากดแล้วจะไปทำงานที่ไฟล์ vote.php

ผลลัพธ์

ลักษณะงานแบบใดที่คุณต้องการ	
☐	พนักงานขาย
☐	ออกแบบเว็บ
☒	โปรแกรมเมอร์
Vote View	

รูปที่ 9-4 : แสดงการเลือกโหวต

รักวัวต้องผูก

รักลูกต้องดี

รักมีต้องคำ

รักหน้าต้องคิด

รักมิตรต้องเตือนกัน

ตัวอย่างที่ 9-5 ไฟล์ Vote.php

```

1 <table width="400" border="1" align="center" bordercolor="#FFCCCC">
2 <? include("connect.php");
3 $sqlQ = "select * from poll_question order by q_id desc";
4 $queryQ = mysql_query($sqlQ) or die("error=$sqlQ");
5 $rowQ = mysql_fetch_array($queryQ);
6 $q_id = $rowQ['q_id'];
7 $q_detail = $rowQ['q_detail'];
8 if(isset($_POST['a_id']))
9 {
10     $sqlU = "update poll_answer set a_score=a_score+1 where a_id='{$_POST['a_id']}'";
11     mysql_query($sqlU) or die("error=$sqlU");
12 } ?>
13 <tr><td colspan="3" bgcolor="#FFE1E1"><?=$q_detail?></td></tr>
14 <?
15     $sqlS = "select sum(a_score) from poll_answer where q_id=$q_id";
16     $queryS = mysql_query($sqlS) or die("error=$sqlS");
17     $rowS = mysql_fetch_array($queryS);
18     $sqlA = "select * from poll_answer where q_id=$q_id";
19     $queryA = mysql_query($sqlA) or die("error=$sqlA");
20     $numA = mysql_num_rows($queryA);
21     for($i=1;$i<=$numA;$i++) {
22         $rowA = mysql_fetch_array($queryA);
23         $per = ($rowA['a_score']*100)/$rowS[0]; ?>
24 <tr><td width="100" bgcolor="#FFBFBF"><?=$rowA['a_detail']?></td>
25 <td width="266" bgcolor="#FFBFBF">
26 <hr align="left" color="#333333" size="10" width="<?=$per?>%> </td>
27 <td width="67" bgcolor="#FFBFBF"><?=number_format($per,2)?>%</td>
28 </tr> <? } ?>
29 <tr>
30 <td colspan="3" align="center" bgcolor="#FFE1E1">
31 <input type="button" name="Submit2" value="BaCk" onClick="window.location='index.php';"> </td>
32 </tr>
33 </table>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 2-7 เป็นคำสั่ง เพื่อดึงคำถามสุดท้ายออกมาจากฐานข้อมูล

บรรทัดที่ 8 เป็นคำสั่งเพื่อตรวจสอบ ว่าได้คลิกคำตอบแล้ว

บรรทัดที่ 10 เป็นคำสั่ง SQL เพื่อให้ปรับปรุง ค่าคำตอบที่เลือกจากฟอร์ม ก่อนหน้า เพิ่มค่าอีก

1 ในฐานข้อมูล ตาราง poll_answer

บรรทัดที่ 13 เป็นคำสั่งแสดงคำถาม ที่ได้มาจากฐานข้อมูล

บรรทัดที่ 15-17 เป็นคำสั่ง เพื่อดึงผลรวมของคำตอบของคำถามนี้ ทั้งหมดมาจากฐานข้อมูล

บรรทัดที่ 18-20 เป็นคำสั่ง เพื่อดึงคำตอบของคำถามนี้ มาจากฐานข้อมูล

บรรทัดที่ 21-22 เป็นคำสั่ง เพื่อแสดงคำตอบทีละคำตอบ ของคำถามนี้ โดยใช้ลูป for

บรรทัดที่ 23 เป็นคำสั่ง คำนวณ เปอร์เซนต์ ของคำตอบนี้ ว่ามีค่าเท่าไร โดยหารด้วยผลรวมทั้งหมด คูณด้วย 100

บรรทัดที่ 24 เป็นคำสั่ง เพื่อแสดงรายละเอียดคำตอบ

บรรทัดที่ 26 เป็นคำสั่ง แสดง HR โดยกำหนดความกว้างเป็น เปอร์เซนต์ที่คำนวณได้

บรรทัดที่ 27 เป็นคำสั่ง เพื่อแสดงจำนวนเปอร์เซนต์ เป็นทศนิยม 2 ตำแหน่ง

บรรทัดที่ 31 เป็นคำสั่ง สร้างปุ่มเพื่อกดปุ่ม Back แล้วไปทำงานที่ไฟล์ index.php ผลลัพธ์

ลักษณะงานแบบใดที่คุณต้องการ		
พนักงานขาย	☐	17.65%
ออกแบบเว็บ	☐	23.53%
โปรแกรมเมอร์	☐	58.82%
BaCk		

รูปที่ 9-5 : แสดงผลการโหวต

การสร้างห้องสนทนา

การสร้างห้องสนทนา ไม่มีฐานข้อมูล มีเพียงไฟล์ message เพื่อเก็บข้อความสนทนา โดยมีไฟล์ที่เกี่ยวข้องเพียง 5 ไฟล์ คือ ไฟล์ Index.HTML , ไฟล์ frame.php , ไฟล์ message.html , ไฟล์ chatform.php และไฟล์ chat.php โดยการทำงานเริ่มจากไฟล์ Index.html เพื่อรับข้อมูลชื่อ หรือ Log In เข้าห้องสนทนา จากนั้นไปทำงานที่ไฟล์ frame.php เพื่อแสดงข้อมูลจากไฟล์ Message.HTML และ ไฟล์ ChatForm.php เมื่อมีการพิมพ์ข้อมูล จะส่งให้ไฟล์ chat.php เก็บข้อมูลลงไฟล์ message.html ให้ โดยไฟล์ message.html จะมีการ refresh ทุก 3 วินาที เพื่อข้อมูลเป็นปัจจุบัน หรือ ช้า เร็ว สามารถกำหนดได้

ตัวอย่างที่ 9-6 ไฟล์ INDEX.HTML

```

1 <center>
2 <FORM METHOD=POST ACTION="frame.php">
3 Name :: <INPUT TYPE="text" NAME="txtUser"><br>
4 <INPUT TYPE="submit" value = "go to chat">
5 </FORM>
▶6 </center>

```

อธิบายรหัสคำสั่ง

- บรรทัดที่ 1 เป็นคำสั่ง HTML เพื่อกำหนดให้ข้อมูลอยู่กึ่งกลางหน้า
- บรรทัดที่ 2 เป็นการสร้างฟอร์ม กำหนดให้กดปุ่ม แล้วไปทำงานที่ไฟล์ Frame.php
- บรรทัดที่ 3 เป็นคำสั่งเพื่อแสดง Name แล้วมีกล่องข้อความ รอรับ ชื่อผู้ใช้
- บรรทัดที่ 4 เป็นคำสั่งเพื่อสร้างปุ่ม Submit และแสดงข้อความ Go to chat

ผลลัพธ์

Name ::

ตัวอย่างที่ 9-7 ไฟล์ Frame.php

```

1  <? ob_start();
2    session_start();
3    $username = $_POST['txtUser'];
4    session_register("username");  ?>
5  <? ob_end_flush(); ?>
6  <FRAMESET ROWS="60%,*" >
7    <FRAME SRC="message.html" NAME="">
8    <FRAME SRC="chatform.php" NAME="">
9  </FRAMESET>

```

อธิบายรหัสคำสั่ง

- บรรทัดที่ 1 เป็นคำสั่งเพื่อกำหนดให้ส่งข้อมูลพร้อมกันทีเดียว
- บรรทัดที่ 2 เป็นคำสั่งให้เริ่มมีการทำงานตัวแปร Session
- บรรทัดที่ 3 เป็นคำสั่งเพื่อดึงข้อมูล ที่กรอกจากฟอร์ม มาเก็บไว้ในตัวแปร Username
- บรรทัดที่ 4 เป็นคำสั่งเพื่อบันทึกตัวแปร Session ชื่อ Username
- บรรทัดที่ 5 เป็นคำสั่งเพื่อล้างบัฟเฟอร์
- บรรทัดที่ 6 เป็นคำสั่งสร้าง Frame เพื่อแสดงข้อมูล 2 ไฟล์ คือ ไฟล์ที่ 1 มีความกว้าง 60 % ที่เหลือให้เป็นไฟล์ ที่ 2
- บรรทัดที่ 7 เป็นคำสั่งเพื่อเปิดไฟล์ Message.html ในส่วนที่ 1
- บรรทัดที่ 8 เป็นคำสั่งเพื่อทำงาน ไฟล์ ChatForm.html ในส่วนที่ 2
- บรรทัดที่ 9 เป็นคำสั่งเพื่อปิดขอบเขตของ Frame

ตัวอย่างที่ 9-8 ไฟล์ Message.HTML

```

1 <Meta HTTP-EQUIV="refresh" Content = "3">
2 Napatsarun :: 19-11-09 13:16:53::<font color="black">สวัสดี</font><br>
▶3 Somchai :: 19-11-09 13:17:53::<font color="blue">HELLO</font><br>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เป็นคำสั่งเพื่อกำหนดให้ มีการ refresh ข้อมูล ทุก ๆ 3 วินาที

บรรทัดที่ 2-3 เป็นคำสั่งข้อความที่ถูกบันทึกโดยไฟล์ Chat.php

ตัวอย่างที่ 9-9 ไฟล์ ChatForm.php

```

1 <?
2 session_start();
3 ?>
4 <body onload="frmchat.txtMessage.focus()">
5 <FORM METHOD=POST ACTION="chat.php" name="frmchat">
6 <?=$_SESSION['username']?> :: <INPUT TYPE="text" NAME="txtMessage">
7
8 <SELECT NAME="txtColor" size=1>
9 <option value="black" style = "background-color:black;" <? if ($_SESSION['color'] == "black") echo "Selected"?> >BLACK
10 <option value="blue" style = "background-color:blue;" <? if ($_SESSION['color'] == "blue") echo "Selected"?>>BLUE
11 <option value="brown" style = "background-color:brown;" <? if ($_SESSION['color'] == "brown") echo "Selected"?>>BROWN
12 </SELECT>
13
14 <INPUT TYPE="submit" Value = "send"><br>
15 <A HREF="#" onclick = "return icon(':dog:');"><img src = "15.jpg" border=0></A>
16 <A HREF="#" onclick = "return icon(':angry:');"><img src = "18.jpg" border=0></A>
17 <A HREF="#" onclick = "return icon(':smile:');"><img src = "15.jpg" border=0></A>
18 </FORM>
19 <script language = "JavaScript">
20     function icon(sth)
21     {
22         frmchat.txtMessage.value = frmchat.txtMessage.value + " " + sth ;
23     }
24 </script>
25

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 2 คำสั่ง session_start() เป็นคำสั่งเพื่อเปิดการทำงานใช้ตัวแปร Session

บรรทัดที่ 4 เป็นคำสั่ง เพื่อกำหนดเมื่อมีการเปิดฟอร์ม ให้รองรับค่าที่ txtMessage

บรรทัดที่ 5 เป็นคำสั่ง เพื่อสร้างฟอร์ม กำหนดชื่อเป็น FrmChat

บรรทัดที่ 6 เป็นคำสั่ง เพื่อแสดงชื่อ ผู้ใช้ ผ่านตัวแปร Session Username และรอรับข้อความ
 บรรทัดที่ 8-12 เป็นคำสั่ง เพื่อแสดงแถบสี เป็น List Box และถ้ามีการเลือกแล้วให้แสดงสีที่
 เลือก

บรรทัดที่ 14-20 เป็นคำสั่ง เพื่อแสดง icon ที่เป็นอารมณ์ เปลี่ยนเป็นข้อความด้วย ฟังก์ชัน
 Icon แล้วต่อในข้อความ txtMessage

ตัวอย่างที่ 9-10 ไฟล์ Chat.php

```

1  <?
2  session_start();
3  error_reporting(E_ALL ^ E_NOTICE);
4  function icon($word)
5  {
6      $sign = Array(":dog:",":angry:",":smile:");
7      $img = Array("15.jpg","18.jpg","15.jpg");
8      for ($i=0;$i<sizeof($sign); $i++)
9      {
10         $word=ereg_replace($sign[$i],"<img src=\"".$img[$i].\">",$word);
11     }
12     return $word;
13 }
14 function check($word)
15 {
16     # -----
17     $word = htmlspecialchars($word);
18     $word = ereg_replace(" ","&nbsp;",$word);
19     $word = nl2br($word);
20     # -----
21     $bad = Array("bad","damn");
22     for ($i=0;$i<SizeOf($bad); $i++)
23     {
24         $word = eregi_replace ($bad[$i],"<FONT COLOR=red>***</FONT>",$word);
25     }
26     return $word;
27 }
```

```

28 #-----TEMP OLD MESSAGE-----
29 $temp = file("message.html");
30 if (sizeof($temp) > 10)
31     $start = sizeof($temp) - 10;
32 for ($i = $start; $i<sizeof($temp); $i++)
33 {
34     $old.=$temp[$i];
35 }
36 #-----
37 $color = $_POST['txtColor'];
38 session_register("color");
39 $_SESSION['color'] = $color;
40 $openfile = fopen("message.html","w");
41 $newmsg = check($_POST['txtMessage']);
42 $newmsg = icon($newmsg);
43 $newmsg = "<font color=\"\$color\">$newmsg</font>";
44 $newmsg = $_SESSION['username'] . " : ". Date("d-m-y H:i:s") . " : ". $newmsg;
45
46 $newmsg .= "<br>\n";
47 $header = "<Meta HTTP-EQUIV=\"refresh\" Content = \"3\">\n";
48 fputs($openfile,$header);
49 fputs($openfile,$old);
50 fputs($openfile,$newmsg);
51 fclose ($openfile);
52 #-----
53 header("Location:chatform.php")
54 ?>

```

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เครื่องหมาย <? หมายถึง เปิดคำสั่ง PHP

บรรทัดที่ 2 คำสั่ง session_start() เป็นคำสั่งเพื่อเปิดการทำงานใช้ตัวแปร Session

บรรทัดที่ 3 เป็นคำสั่งเพื่อกำหนดการแสดงผลผิดพลาด ยกเว้น ไม่แสดงผลผิดพลาดแบบคำเตือน

บรรทัดที่ 4-13 เป็นฟังก์ชัน เพื่อเปลี่ยนจากข้อความอารมณ์ เป็นการแสดงภาพแทน

บรรทัดที่ 14-27 เป็นฟังก์ชัน เพื่อตรวจสอบอักขระพิเศษ เว้นวรรค และคำหยาบ

บรรทัดที่ 17 เป็นคำสั่งเพื่อแปลง เครื่องหมายพิเศษของ tag HTML ให้เป็นรหัสอักขระ ที่ใช้ในภาษา HTML เช่น เป็น เป็นต้น

บรรทัดที่ 18 เป็นคำสั่งเพื่อแทนที่ช่องว่างด้วย รหัส

บรรทัดที่ 19 เป็นคำสั่งเพื่อเปลี่ยน สัญลักษณ์ Enter ด้วย

บรรทัดที่ 21-25 เป็นคำสั่งเพื่อแทนที่คำหยาบด้วย ***

บรรทัดที่ 28-35 เป็นคำสั่งเพื่อเก็บข้อความเก่า 10 ข้อความ

บรรทัดที่ 37-39 เป็นคำสั่งเพื่อค่าสีเป็นตัวแปร Session

บรรทัดที่ 40 เป็นคำสั่งเพื่อเปิดไฟล์ Message เพื่อเขียนใหม่ ลบของเก่าทิ้ง

บรรทัดที่ 41 เป็นคำสั่งเพื่อเช็คข้อความ

บรรทัดที่ 42 เป็นคำสั่งเพื่อเปลี่ยนข้อความเป็นแสดงรูป

บรรทัดที่ 43 เป็นคำสั่งเพื่อใส่สี

บรรทัดที่ 44 เป็นคำสั่งเพื่อแสดงค่า วันที่ ในตัวแปร newmsg

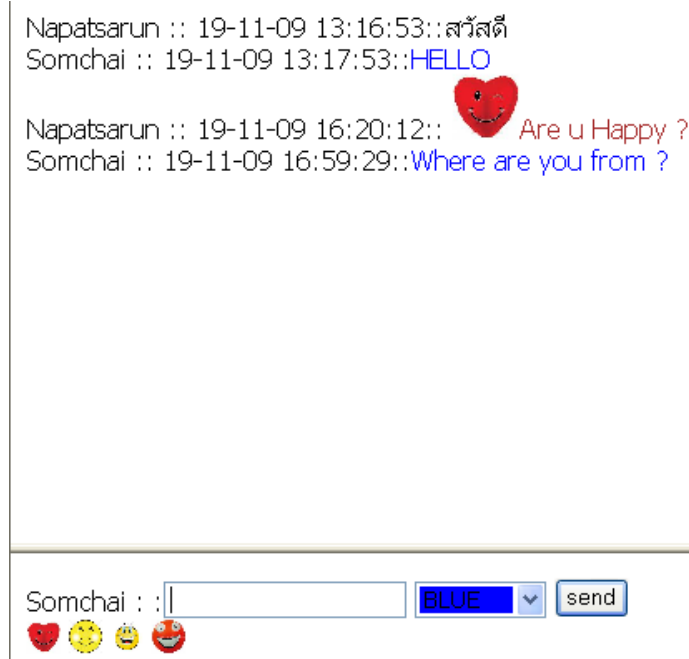
บรรทัดที่ 47 เป็นคำสั่งเพื่อเก็บค่า Header เป็นคำสั่งที่ใช้ในการ Refresh

บรรทัดที่ 48-50 เป็นคำสั่ง เขียนไฟล์ Message ด้วยค่า Header ข้อความเก่า และข้อความใหม่

บรรทัดที่ 51 เป็นคำสั่งปิดไฟล์

บรรทัดที่ 53 เป็นคำสั่งให้กลับไปฟอร์มรับข้อมูลมาใหม่

ผลลัพธ์



รูปที่ 9-6 : แสดงผลการสนทนา

การนับจำนวนผู้เข้าชม

การนับจำนวนผู้ชมไม่มีฐานข้อมูลเก็บ จะเก็บจำนวนเก่าไว้ในไฟล์ Counter.txt แล้วไฟล์ Counter.php จะอ่านไฟล์ ขึ้นมาแล้วบวกเพิ่มไปเรื่อย ๆ

อย่าอยู่ว่าง

อย่าห่างผู้ใหญ่

จะหลงทางได้ง่าย

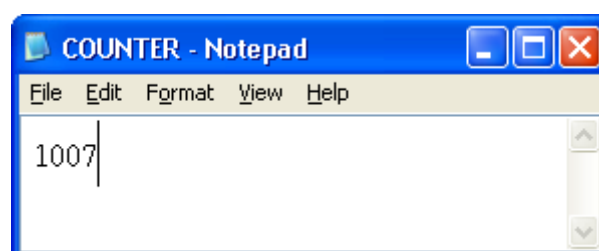
ตัวอย่างที่ 9-11 ไฟล์ Counter.php

```

1  <?
2  error_reporting(E_ALL ^ E_NOTICE);
3  #counter.php
4  #—— read text File ——
5  $openfile = fopen("counter.txt","r");
6  $counter = fread($openfile,10);
7  fclose ($openfile);
8  #—————
9  if (!isset($_COOKIE['visited']))
10 {
11     setcookie("visited","yes",time()+86400);
12     echo $_COOKIE['visited'];
13     $counter++;
14 }
15 #—— write text File ——
16 $openfile = fopen("counter.txt","w");
17 fputs($openfile,$counter);
18 fclose ($openfile);
19 echo "You are # ".$counter;
20 #— begin display counter —
21 if (strlen($counter) < 6)
22 {
23     $rem = 6 - strlen($counter);
24     for ($i=1;$i<=$rem;$i++)
25         echo "<img src='0.gif'>";
26 }
27 $num=0;
28 for ($i=0;$i<strlen($counter);$i++)
29 {
30     $num = $counter{$i};
31     echo "<img src='$num.gif'>";
32 }
33 ?>

```

ตัวอย่างที่ 9-12 ไฟล์ Counter.txt



รูปที่ 9-7 : แสดงไฟล์ Counter.txt

อธิบายรหัสคำสั่ง

บรรทัดที่ 1 เครื่องหมาย <? หมายถึง เปิดคำสั่ง PHP

บรรทัดที่ 2 เป็นคำสั่งเพื่อกำหนดการแสดงผลของข้อมูลที่ได้จากฟังก์ชัน rand() ไม่แสดงข้อมูลในรูปแบบคำเตือน

บรรทัดที่ 5 เป็นคำสั่งเปิดไฟล์ counter.txt เพื่ออ่าน

บรรทัดที่ 6 เป็นคำสั่งอ่านข้อมูลจากไฟล์ มา 10 ตัว

บรรทัดที่ 7 เป็นคำสั่งปิดไฟล์ที่เปิดมา

บรรทัดที่ 9-13 เป็นคำสั่งตรวจสอบค่า Cookie ว่าเคยเข้ามาเกิน 1 วันแล้วให้บวกค่า counter

อีก 1 บรรทัดที่ 16 เป็นคำสั่งเปิดไฟล์ counter.txt เพื่อเขียนใหม่

บรรทัดที่ 17 เป็นคำสั่งเพื่อเขียนค่า counter ใหม่ลงไฟล์ counter.txt

บรรทัดที่ 18 เป็นคำสั่งเพื่อปิดไฟล์

บรรทัดที่ 19 เป็นคำสั่งเพื่อแสดงจำนวน counter

บรรทัดที่ 21-26 เป็นคำสั่งเพื่อตรวจสอบว่าค่าน้อยกว่า 6 หลักให้แสดง 0 ข้างหน้า โดยใช้ไฟล์รูป 0

บรรทัดที่ 27-32 เป็นคำสั่งเพื่อรูปภาพ ของตัวเลขนั้น ๆ

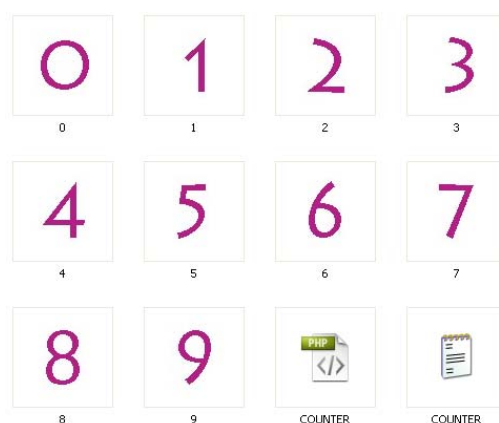
ผลลัพธ์

You are # 1008



รูปที่ 9-8 : แสดงผลลัพธ์การนับจำนวนผู้ชม

ข้อมูลไฟล์ ที่ใช้แสดงภาพ



รูปที่ 9-9 : แสดงไฟล์ภาพในการแสดงตัวเลข

รากฐานของสังคมไทยเป็นสังคมแบบเกษตรกรรม ข้าว คือสายใยชีวิตที่สร้างความอยู่เย็นเป็นสุขมาช้านาน

“ข้าพเจ้ามีโอกาสได้ศึกษาและทดลองทำนาบ้าง และทราบดีว่าการทำนา นั้นมีความยากลำบาก และเป็นอุปสรรคอยู่มากน้อย..”

พระราชปณิธานหนึ่งขององค์พระมหากษัตริย์มีอยู่ว่า จะทำอย่างไรให้ชาวนาอยู่ได้ตามอัตภาพ จึงไม่ใช่เรื่องแปลกนักที่เราจะเห็นพระองค์ทรงดำเนินพระราชกรณียกิจ อันเกี่ยวข้องกับการเกษตรอย่างมากมาย และยังได้ทราบว่าในสวนจิตรลดา ซึ่งเป็นเขตพระราชฐานมีทุ่งนาอันเป็นโครงการส่วนพระองค์แล้ว ยิ่งย่ำชัดว่าพระองค์ทรงให้ความสำคัญเพียงไรกับการเกษตรกรรม

ย้อนกลับไปในช่วงปี พุทธศักราช 2479 – 2502 พระราชพิธีจรดพระนังคัลแรกนาขวัญ ได้วางเว้นไป เพียงจัดให้มีพระราชพิธีพืชมงคลเท่านั้น จนกระทั่งในปี พุทธศักราช 2503 พระบาทสมเด็จพระเจ้าอยู่หัวฯ ทรงโปรดให้ฟื้นฟูพระราชพิธี จรดพระนังคัลแรกนาขวัญขึ้นอีก และปรับปรุงให้เหมาะกับยุคสมัย และปฏิบัติสืบมามิได้ขาด ด้วยทรงเห็นว่าเป็นการรักษาพระราชประเพณีอันดีงาม มีผลในการบำรุงขวัญและจิตใจของคนไทย ซึ่งส่วนใหญ่เป็นเกษตรกร

จนถึงวันนี้ เป็นเวลากว่า 40 ปีแล้ว ที่ความรู้ และเมล็ดพันธุ์พระราชทานจากโครงการนาข้าวสาธิต ได้แจกจ่ายไปยังเกษตรกรทั่วประเทศเพื่อเป็นมิ่งขวัญและสิริมงคล ในการประกอบอาชีพเกษตรกรรมต่อไป

จากสภาพความเสื่อมโทรมของทรัพยากรดิน อันเนื่องมาจากฝนตกหนัก และน้ำไหลบ่าอย่างรุนแรง ทำให้ผิวดินถูกกัดเซาะจนสูญเสียความอุดมสมบูรณ์ เป็นเหตุให้ดินพังทลาย ส่งผลเสียหายต่อพื้นที่ทำการเกษตร

พระบาทสมเด็จพระเจ้าอยู่หัวฯ ทรงตระหนักถึงปัญหา และสาเหตุที่เกิดขึ้น จึงทรงศึกษาศักยภาพของ “หญ้าแฝก” ที่มีคุณสมบัติพิเศษในการช่วยป้องกันการชะล้างพังทลายของหน้าดิน และช่วยรักษาความชุ่มชื้นของน้ำใต้ดินไว้

หญ้าแฝก เป็นพืชพื้นบ้านที่ทำหน้าที่เสมือนกำแพงธรรมชาติที่มีชีวิต ช่วยชะลอการไหลของน้ำ และแรงปะทะของลม คอยกักกันตะกอนดิน ไม่ให้หน้าดินพังทลาย ซึ่งเกษตรกรสามารถนำมาปลูกโดยไม่ต้องดูแลมากนัก อีกทั้งยังช่วยประหยัดค่าใช้จ่าย สถาบัน international erosion control association หรือ IECA ได้ทูลเกล้าฯ ทูลกระหม่อมถวายรางวัล international merit award และธนาคารโลก ได้ทูลเกล้าฯ ทูลกระหม่อม ถวายรางวัลรากหญ้าซบสาริต แต่พระบาทสมเด็จพระเจ้าอยู่หัวฯ เพื่อเป็นการสดุดีพระเกียรติคุณ และพระปรีชาสามารถในฐานะที่ทรงเป็นนักอนุรักษ์ดินและน้ำ นับตั้งแต่ พ.ศ. 2534 มาจนถึงปัจจุบัน ผลสำเร็จของโครงการพระราชดำริเรื่อง “หญ้าแฝก” ได้พลิกฟื้นคืนชีวิตให้กับแผ่นดิน นำมาซึ่งความอุดมสมบูรณ์ของระบบนิเวศอีกครั้ง

วันนี้หญ้าแฝกจึงมิได้เป็นเพียงต้นหญ้าที่ไร้ค่า แต่เป็น “ต้นหญ้ามหัศจรรย์” ที่มีเมือกอันดีต่อผืนแผ่นดินไทย

ที่มา :: <http://www.royalvdo.com/?p=11>

บทที่ 10 การประยุกต์ใช้งาน

การประยุกต์ใช้งาน การสร้างโปรแกรมส่วนใหญ่จะมีการส่งเมลเพื่อยืนยันการทำงานจากระบบ ว่าการทำงานสมบูรณ์หรือไม่ การส่งเมลต้องมีเครื่องแม่ข่ายที่จะใช้ในการส่ง โดยมีข้อมูลที่ใช้ในการส่งดังนี้ อีเมลผู้รับ หัวข้อ เนื้อหา อีเมลผู้ส่ง ดูจากตัวอย่าง ดังนี้

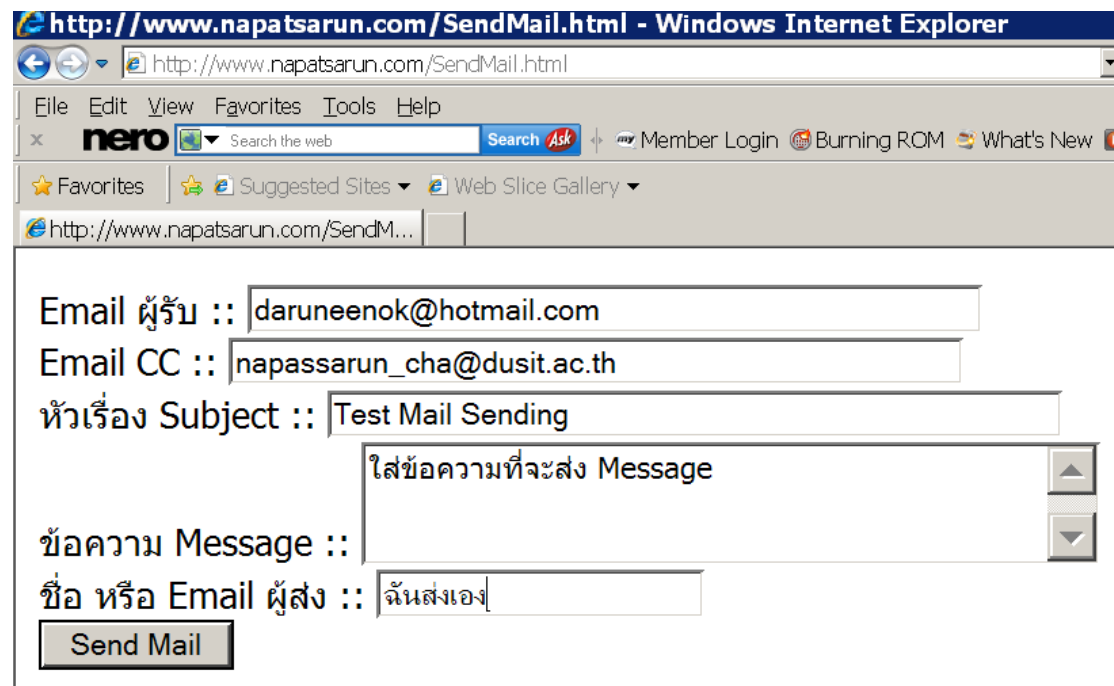
ตัวอย่างที่ 10-1 ไฟล์ frmEmail.html

```
<FORM method=POST ACTION="testMail.php">
Email ผู้รับ :: <Input type="text" name="to" value="daruneenok@hotmail.com"
size=50><BR>
Email CC :: <Input type="text" name="cc"
value="napassarun_cha@dusit.ac.th" size=50><BR>
หัวข้อ Subject :: <Input type="text" name="subject" value="Test Mail Sending"
size=50><BR>
ข้อความ Message :: <textarea name="message" rows=3 cols=50>
ใส่ข้อความที่จะส่ง Message
</textarea><BR>
ชื่อ หรือ Email ผู้ส่ง :: <input type="text" name="from"
value="napatsarunc@yahoo.com"><BR>
<input type="submit" value="Send Mail">
</FORM>
```

ตัวอย่างที่ 10-2 ไฟล์ testMail.php

```
<?php
$to=$_POST['to'];
$subject=$_POST['subject'];
$message=$_POST['message'];
$from=$_POST['from'];
$cc=$_POST['cc'];
print "คุณต้องการส่ง Mail ให้ :: $to <br>";
print "หัวข้อ :: $subject <br>";
print "เนื้อหา :: $message <br>";
print "ผู้ส่ง :: $from <br>";
print "CC :: $cc <br>";
$headers = "From: $from\n";
$headers .= "CC: $cc\n";
print "HEADER :: $headers <BR>";
if (mail($to,$subject,$message,$headers))
    echo "ส่งอีเมลเรียบร้อยแล้ว.";
else
    echo "ส่งอีเมลไม่ได้.";
?>
```

ผลลัพธ์ ไฟล์ frmEmail.html



http://www.napatsarun.com/SendMail.html - Windows Internet Explorer

http://www.napatsarun.com/SendMail.html

File Edit View Favorites Tools Help

Search the web Search Ask Member Login Burning ROM What's New

Favorites Suggested Sites Web Slice Gallery

http://www.napatsarun.com/SendM...

Email ผู้รับ :: daruneenok@hotmail.com

Email CC :: napassarun_cha@dusit.ac.th

หัวข้อ Subject :: Test Mail Sending

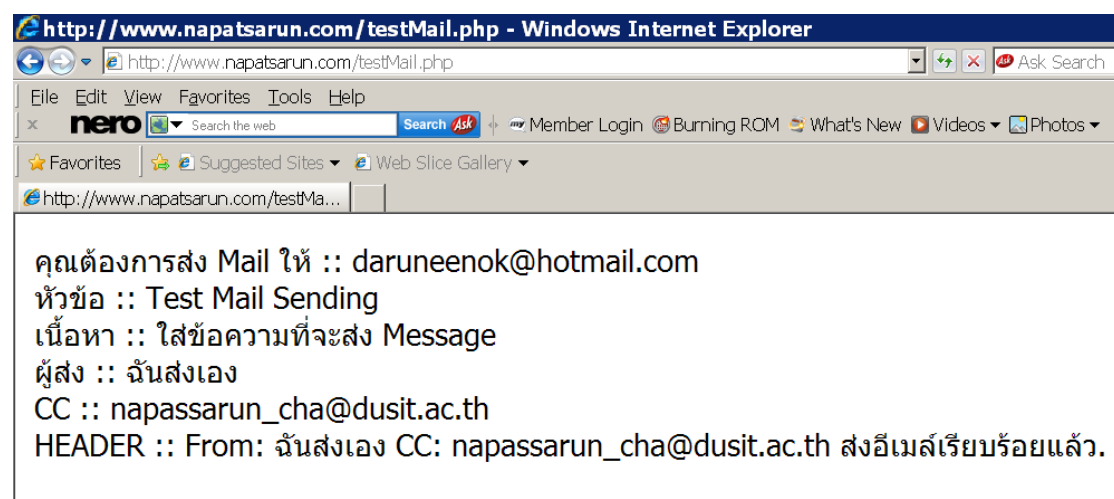
ใส่ข้อความที่จะส่ง Message

ข้อความ Message ::

ชื่อ หรือ Email ผู้ส่ง :: ฉันส่งเอง

Send Mail

ผลลัพธ์ ไฟล์ testMail.php



http://www.napatsarun.com/testMail.php - Windows Internet Explorer

http://www.napatsarun.com/testMail.php

File Edit View Favorites Tools Help

Search the web Search Ask Member Login Burning ROM What's New Videos Photos

Favorites Suggested Sites Web Slice Gallery

http://www.napatsarun.com/testMa...

คุณต้องการส่ง Mail ให้ :: daruneenok@hotmail.com

หัวข้อ :: Test Mail Sending

เนื้อหา :: ใส่ข้อความที่จะส่ง Message

ผู้ส่ง :: ฉันส่งเอง

CC :: napassarun_cha@dusit.ac.th

HEADER :: From: ฉันส่งเอง CC: napassarun_cha@dusit.ac.th ส่งอีเมลเรียบร้อยแล้ว.

ผลลัพธ์ การได้รับ Email

Current Folder: **INBOX****Welcome:** napassarun_cha[Select All](#)

Move Selected To:

INBOX [Move](#)[Forward](#)[Flag](#)[Unflag](#)

	From	Date	Subject
☐	ฉันส่งเอง @cat247188.lnwhoostname.com	11:45 am	Test Mail Sending

Subject: Test Mail Sending**From:** ฉันส่งเอง@cat247188.lnwhoostname.com**Date:** Sat, February 12, 2011 11:45 am**To:** daruneenok@hotmail.com**Cc:** napassarun_cha@dusit.ac.th**Priority:** Normal**Options:** [View Full Header](#) | [View Printable Version](#)

ใส่ข้อความที่จะส่ง Message

บรรณานุกรม

ดวงกมล กลีบจินดา และ กิตติเชษฐ ยี่งกัฒไพบูลย์ , “การพัฒนาเว็บด้วย JavaScript , CSS และ DOM” , พิมพ์ครั้งที่ 2 , สำนักพิมพ์เคทีพี , 2551.

วิชา ศิริธรรมจักร , “Web Programming ด้วย AJAX และ PHP” , พิมพ์ครั้งที่ 1 , สำนักพิมพ์เคทีพี , 2549.

พิรพร หมุนสนธิ และ วันวิสาข์ เมฆฉาย , “ใช้งาน JavaScript แบบมืออาชีพ” , พิมพ์ครั้งที่ 2 , สำนักพิมพ์เคทีพี , 2551.

ธีระศักดิ์ สุโชตินันท์ , “ทำไป ลองไป จาวาสคริปส์ ” , พิมพ์ครั้งที่ 1 , สำนักพิมพ์สวัสดิ์ ไอที, 2550.